Building Android Apps with HTML, CSS, and JavaScript

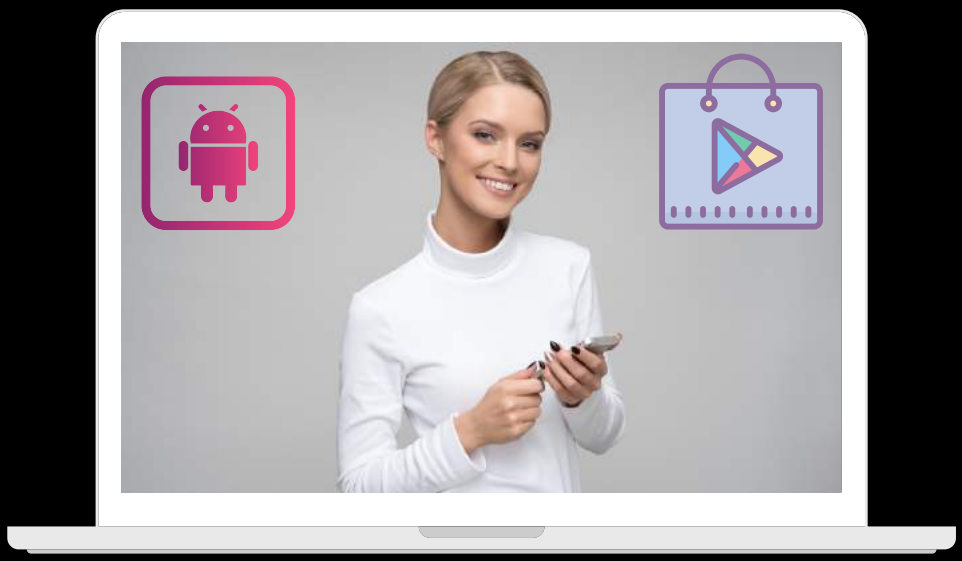
IN HINDI

Machhindra kalan

DIGITAL DOWNLOAD हिंदी

Building Android Apps with HTML, CSS, and JavaScript

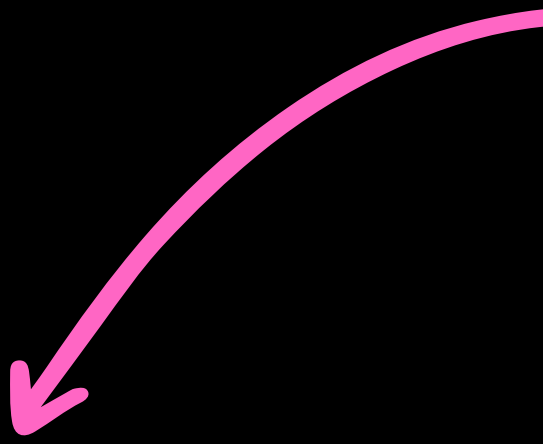
Thanks for purchasing this Digital Download.



What's the success?

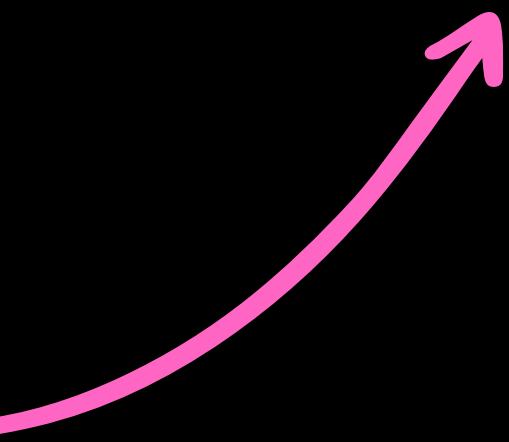
“We learn from failure, not from success!”
— Bram Stoker, Dracula

Please Keep this PDF in a safe place!



SUBSCRIBE TO FREE EBOOKS

www.cprogrammingbookinhindi.com



Building Android Apps with HTML, CSS, and JavaScript in Hindi

Copyright © 2017 by Machhindra Kalan

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Lead Editor: Machhindra Kalan

Distributed to the book trade worldwide by Web Study Lab Pvt Ltd, Location - Mumbai, Palava city Pin 421204

E-mail: machhi29@gmail.com, or

Visit: www.cprogrammingbookinhindi.com

For information on translations, please contact Web Study Lab Pvt Ltd, Location - Mumbai, Palava city Pin 421204

Phone +91 86555-67778

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, the author shall not have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this book.

This book
is dedicated
to those

who really wants
to be a

Building Android Apps with HTML, CSS, and JavaScript In Hindi

INDEX OF CONTENTS

Table of Contents

1. Introduction 1
Web Apps Vs Native Apps 1
What Is a Web App? 1
What Is a Native App? 1
Pros and Cons 2
Which Approach Is Right for You? 2
Web Programming Crash Course 3
Introduction to HTML 3
Introduction to CSS 6
Introduction to JavaScript 9

2. Basic Styling 13
Don't Have a Website? 13
First Steps 14
Prepare a Separate Android Stylesheet 17
Control the Page Scaling 19
Adding the Android CSS 20
Adding the Android Look and Feel 23
Adding Basic Behavior with jQuery 25
What You've Learned 31

3. Advanced Styling 33
Adding a Touch of Ajax 33
Traffic Cop 33
Setting Up Some Content to Work With 36
Routing Requests with JavaScript 36
Simple Bells and Whistles 38
Progress Indicator 38
Setting the Page Title 41

Handling Long Titles 43
Automatic Scroll-to-Top 44
Hijacking Local Links Only 45
Roll Your Own Back Button 46
Adding an Icon to the Home Screen 52
What You've Learned 53

4. Animation 55
With a Little Help from Our Friend 55
Sliding Home 55
Adding the Dates Panel 58
Adding the Date Panel 60
Adding the New Entry Panel 62
Adding the Settings Panel 64
Putting It All Together 66
Customizing jQTouch 68
What You've Learned 70

5. Client-Side Data Storage 71
Web Storage 71
Saving User Settings to Local Storage 72
Saving the Selected Date to Session Storage 75
Web SQL Database 76
Creating a Database 78
Inserting Rows 80
Selecting Rows and Handling Result Sets 84
Deleting Rows 88
What You've Learned 91
Web Database Error Code Reference 91

6. Going Offline 93
The Basics of the Offline Application Cache 93
Online Whitelist and Fallback Options 96
Creating a Dynamic Manifest File 99
Debugging 105
The JavaScript Console 106
What You've Learned 108

7. Going Native 109
Introduction to PhoneGap 109
Download the Android SDK 110
Download PhoneGap 114
Setting Up the Environment 115

Create an Android Virtual Device 117
Build KiloGap 118
Installing KiloGap in the Emulator 122
Using the Screen's Full Height 125
Customizing the App Icon 126
Installing KiloGap on Your Phone 127
Controlling the Phone with JavaScript 128
Beep, Vibrate, and Alert 128
Geolocation 132
Accelerometer 137
What You've Learned 140

8. Submitting Your App to the Android Market..... 141
Preparing a Release Version of Your App 141
Removing Debug Code 141
Versioning Your App 142
Signing Your App 142
Uploading Your App to the Android Market 146
Distributing Your App Directly 147
Further Reading 148

INTRODUCTION

इससे पहले कि हम शुरू करे, मैं जल्दी से खेल का मैदान स्थापित करना चाहूँगा। इस अध्याय में, मैं मुख्य शब्दों को परिभाषित करूंगा, दो सबसे सामान्य विकास एप के प्रोसैस और विपक्ष की तुलना करूंगा, और इस पुस्तक में उपयोग की जाने वाली तीन मुख्य वेब तकनीकों पर क्रैश कोर्स दे सकता हूं।

Web Apps Vs Native Apps

सबसे पहले, मैं वेब ऐप और नेटिव ऐप से क्या मतलब रखता हूं और उनके पेशेवरों और विपक्षों पर विचार करना चाहता हूं।

What Is a Web App?

मेरे लिए, एक वेब ऐप मूल रूप से एक वेबसाइट है जो विशेष रूप से स्मार्टफोन पर उपयोग के लिए अनुकूलित है। साइट सामग्री एक मानक छोटे व्यवसाय ब्रोशर साइट से एक बंधक कैलकुलेटर से दैनिक कैलोरी ट्रैकर तक कुछ भी हो सकती है – सामग्री अप्रासंगिक है। एक वेब ऐप की परिभाषित विशेषताएं यह हैं कि उपयोगकर्ता इंटरफ़ेस (UI) वेब मानक प्रौद्योगिकियों के साथ बनाया गया है, यह एक URL (सार्वजनिक, निजी, या शायद एक लॉगिन के पीछे) पर उपलब्ध है, और यह एक मोबाइल डिवाइस की विशेषताओं के लिए अनुकूलित है। फोन पर एक वेब ऐप इंस्टॉल नहीं किया गया है, यह एंड्रॉइड मार्केट में उपलब्ध नहीं है, और यह जावा के साथ नहीं लिखा गया है।

What Is a Native App?

इसके विपरीत, मूल एपलिकेशन एंड्रॉइड फोन पर इंस्टॉल किए जाते हैं, उनके पास हार्डवेयर (स्पीकर, एक्सेलेरोमीटर, कैमरा, आदि) तक पहुंच होती है, और वे जावा के साथ लिखे जाते हैं। हालांकि, एक native ऐप की परिभाषित विशेषता यह है कि यह एंड्रॉइड मार्केट में

उपलब्ध है– एक ऐसी सुविधा जिसने दुनिया भर में सॉफ्टवेयर एंट्रेप्रेन्यूरो के एक गिरोह की कल्पना को पकड़ लिया है, मुझे शामिल किया गया है।

Pros and Cons

विभिन्न applications की अलग–अलग आवश्यकताएं हैं। कुछ ऐप्स दूसरों की तुलना में वेब तकनीकों के साथ बेहतर फिट हैं। प्रत्येक दृष्टिकोण के पेशेवरों और विपक्षों को जानने से आपको बेहतर निर्णय लेने में मदद मिलेगी कि आपकी स्थिति के लिए कौन सा मार्ग उपयुक्त है।

Here are the pros of native app development:

- लाखों पंजीकृत क्रेडिट कार्ड मालिक एक क्लिक दूर हैं।
- आप डिवाइस के सभी cool हार्डवेयर सुविधाओं तक पहुंच सकते हैं।

Here are the cons of native app development:

- Android डेवलपर बनने के लिए आपको भुगतान करना होगा
- आपका ऐप केवल एंड्रॉइड फोन पर चलेगा
- आपको जावा का उपयोग करके विकसित करना होगा विकास चक्र धीमा है (विकास, संकलन, परिनियोजन, दोहराएं)

Here are the pros of web app development:

- वेब डेवलपर अपने वर्तमान **(current authoring tools)** संलेखन उपकरण का उपयोग कर सकते हैं
- आप अपने वर्तमान वेब डिज़ाइन और विकास कौशल का

उपयोग कर सकते हैं

- आपका ऐप किसी भी डिवाइस पर चलेगा जिसमें एक वेब ब्राउज़र है
- आप वास्तविक समय में बग को ठीक कर सकते हैं
- विकास चक्र तेज है

Here are the cons of web app development:

- आप फ़ोन की सभी cool हार्डवेयर सुविधाओं तक नहीं पहुँच सकते
- यदि आप ऐप के लिए शुल्क लेना चाहते हैं तो आपको अपना भुगतान सिस्टम रोल करना होगा
- परिष्कृत UI प्रभाव को प्राप्त करना कठनि हो सकता है

Which Approach Is Right for You?

यहां वह रोमांचक है। एंड्रॉइड फोन की हमेशा–ऑनलाइन प्रकृति एक वातावरण बनाती है जिसमें एक वेब ऐप और एक Native ऐप के बीच की रेखाएं धुंधली हो जाती हैं। एंड्रॉइड वेब ब्राउज़र की कुछ छोटी ज्ञात विशेषताएं भी हैं (देखें अध्याय 6) यदि आप चाहें तो आपको एक वेब ऐप ऑफलाइन लेने की अनुमति देते हैं। क्या अधिक है, कई तृतीय–पक्ष परियोजनाएं जिनमें से PhoneGap सबसे उल्लेखनीय है– सक्रिय रूप से समाधान विकसित कर रहे हैं जो वेब डेवलपर्स को एक वेब ऐप लेने और इसे एंड्रॉइड और अन्य मोबाइल प्लेटफार्मों के लिए एक मूल ऐप के रूप में पैकेज करने की अनुमति देते हैं।

मेरे लिए, यह सही मिश्रण है। मैं अपनी मूल भाषा में लिख सकता हूं, एक उत्पाद को एक शुद्ध वेब ऐप के रूप में जारी कर सकता हूं (एंड्रॉइड और किसी भी अन्य डिवाइस के लिए जो एक आधुनिक ब्राउज़र है), और एक समान कोड बेस का उपयोग करके एक बढ़ाया

native version बना सकते हैं जो डिवाइस हार्डवेयर और संभावित रूप से एक्सेस कर सकता है Android Market में बेचा जा सकता है। यह एक शानदार तरीका है

अपने ऐप के लिए "प्रीमियम" मॉडल बनाएं—वेब ऐप पर मुफ्त पहुंच की अनुमति दें और अधिक सुविधा संपन्न native version के लिए शुल्क लें।

Web Programming Crash Course

वेब ऐप्स बनाने के लिए हम जिन तीन मुख्य तकनीकों का उपयोग करेंगे, वे हैं HTML, CSS और Java Script हम यह सुनिश्चित करने के लिए प्रत्येक को जल्दी से कवर करेंगे कि हम फैंसी सामान में जुताई से पहले एक ही पृष्ठ पर हैं।

Introduction to HTML

जब आप वेब ब्राउज़ कर रहे होते हैं, तो जो पृष्ठ आप देख रहे हैं, वे केवल किसी अन्य व्यक्ति के कंप्यूटर पर बैठे टेक्स्ट दस्तावेज़ हैं। एक विशिष्ट वेब पेज का टेक्स्ट HTML टैग में लिपटा होता है, जो आपके ब्राउज़र को दस्तावेज़ की संरचना के बारे में बताता है। इस सूचना के साथ, ब्राउज़र यह तय कर सकता है कि जानकारी को किस तरह से प्रदर्शित किया जाए।

उदाहरण में दिखाए गए वेब पेज स्निपेट पर विचार करें पहली पंक्ति में, वहाँ स्ट्र गि हाय! h1 टैग की एक जोड़ी में लिपटा है। ध्यान दें कि ओपन टैग और क्लोज टैग थोड़ा अलग है: क्लोज टैग में दूसरे वर्ण के रूप में एक स्लैश (/) है, जबकि ओपन टैग में स्लैश नहीं ळें

H1 टैग में रैपिंग टेक्स्ट ब्राउज़र को बताता है कि संलग्न शब्द एक

हेडिंग हैं, जो इसे अपनी ही लाइन पर बड़े बोल्ड टेक्स्ट में प्रदर्शित करेगा। इसमें h2, h3, h4, h5 और h6 हेडिंग टैग भी हैं। संख्या जितनी कम होगी, हेडर उतना ही महत्वपूर्ण होगा, इसलिए h3 टैग में लिपटा गया पाठ, h3 टैग में लिपटे पाठ की तुलना में छोटा (यानी कम महत्वपूर्ण दिखने वाला) होगा।

उदाहरण 1–1 में h1 टैग के बाद, p टैग में दो लाइनें लिपटी हुई हैं। इन्हें पैराग्राफ टैग कहा जाता है। ब्राउजर प्रत्येक पैराग्राफ को अपनी लाइन पर प्रदर्शित करेगा। यदि अनुच्छेद ब्राउज़र विंडो की चौड़ाई को पार करने के लिए पैराग्राफ काफी लंबा है, तो पाठ नीचे झुक जाएगा और अगली पंक्ति पर जारी रहेगा। या तो मामले में, पृष्ठ पर अगले आइटम से अलग करने के लिए पैराग्राफ के बाद एक रिक्त लाइन डाली जाएगी।

Example 1-1. HTML snippet

```
<h1>Hi there!</h1>
<p>Thanks for visiting my web page</p> <p>I hope you like it</p>
```

आप अन्य HTML टैग के अंदर भी HTML टैग लगा सकते हैं। उदाहरण 1–2 एक unordered list (ul) टैग दिखाता है जिसमें तीन सूची आइटम (li) हैं। एक ब्राउज़र में, यह अपनी लाइन पर प्रत्येक आइटम के साथ बुलेटेड सूची के रूप में दिखाई देता है। जब आपके पास एक टैग या टैग किसी अन्य टैग के अंदर होता है, तो आंतरिक टैग को चाइल्ड एलीमेंट्स या पैरेंट्स टैग के बच्चे कहा जाता है। तो इस उदाहरण में, li tags are children of the ul parent हैं।

Example 1-2. Unordered list

```
<ul>
  <li>Pizza</li>
  <li>Beer</li>
  <li>Dogs</li>
</ul>
```

अब तक कवर किए गए टैग सभी ब्लॉक टैग हैं। ब्लॉक टैग की परिभाषित विशेषता यह है कि वे अपनी खुद की एक पंक्ति पर प्रदर्शित होते हैं, जिनमें कोई भी element बाएं या दाएं नहीं होता है। यही कारण है कि title, पैराग्राफ और सूची आइटम पृष्ठ के बजाय नीचे प्रगति करते हैं। एक ब्लॉक टैग के विपरीत एक इनलाइन टैग है, जो कि जैसा कि नाम से ही स्पष्ट है, एक पंक्ति में दिखाई दे सकता है। जोर टैग (em) एक इनलाइन टैग का एक उदाहरण है, और यह इस तरह दिखता है:

```
<p> <em>really</em> hope you like it.</p>
```

इनलाइन टैग्स के ग्रैंडडैडी– और यकीनन HTML की सबसे अच्छी सुविधा है– एक टैग है। "a" anchor के लिए खड़ा है, लेकिन कई बार मैं इसे एक लिंक या हाइपरलिंक के रूप में भी देखूंगा। एंकर टैग में लिपटा हुआ पाठ क्लिक करने योग्य है, जैसे कि उस पर क्लिक करने से ब्राउज़र एक नया HTML पेज लोड करता है।

ब्राउजर को यह बताने के लिए कि कौन सा नया पेज लोड करना है, हमें जोड़ना होगा कि टैग को एट्रीब्यूट–ओट कहा जाता है। विशेषताएँ नाम के मान हैं जिन्हें आप एक खुले टैग में सममिलित करते हैं। एंकर टैग में, आप लक्ष्य पृष्ठ का स्थान निर्दिष्ट करने के लिए href विशेषता का उपयोग करते हैं। यहां Google के होम पेज का लिंक दिया गया है:

```
<a href="http://www.google.com/">Google</a>
```

यदि आप HTML पढ़ने के आदी नहीं हैं, तो यह कुछ हद तक गड़बड़ी जैसा लग सकता है, लेकिन आपको Google होम पेज के लिए URL निकालने में सक्षम होना चाहिए। आप पूरी पुस्तक में बहुत सारे टैग और href विशेषताएँ देख रहे होंगे, इसलिए यदि यह पहली

नज़र में समझ में न आए, तो अपने head को इसके चारों ओर लाने के लिए एक मिनट का समय लें।

विशेषताओं के बारे में ध्यान रखने वाली कुछ बातें हैं। Dif– फ़ेरेंट HTML टैग्स विभिन्न विशेषताओं की अनुमति देते हैं। आप कई जोड़ सकते हैं

रिक्त स्थान के साथ उन्हें अलग करके एक खुले टैग की विशेषता। आप कभी भी किसी बंद टैग में विशेषता नहीं जोड़ते हैं। सैकड़ों संभावित संयोजन हैं विशेषताएँ और टैग, लेकिन यह पसीना नहीं है – हम केवल इस पूरी किताब में एक दर्जन या तो के बारे में चिंता करने की है।

जिस HTML स्निपेट को हम देख रहे हैं, वह सामान्य रूप से संपूर्ण HTML दस्तावेज़ के मुख्य भाग में रहेगा। एक HTML दस्तावेज़ दो classों से बना है: head और body। body वह जगह है जहाँ आप वह सारी सामग्री डालते हैं जो आप चाहते हैं कि उपयोगकर्ता देखें। head में पृष्ठ के बारे में जानकारी होती है, जिसमें से अधिकांश उपयोगकर्ता के लिए अदृश्य है।

body और head हमेशा एक html element में लिपटे रहते हैं। उदाहरण 1–3 एक उचित HTML दस्तावेज़ के संदर्भ में स्निपेट दिखाता है। अभी के लिए हेड सेक्शन कोन एक टाइटल एलिमेंट है, जो ब्राउजर को बताता है कि विंडो के टाइटल बार में कौन सा टेक्स्ट दिखाना है।

Example 1-3. A proper HTML document

Open a text editor and enter the code in Example 1-3

```
<html>
  <head>
<title>My Awesome Page</title>    </head>
  <body>
      <h1>Hi there!</h1>
      <p>Thanks for visiting my web page.</p> <p>I hope you
      like it.</p>
      <ul>
      <li>Pizza</li> <li>Beer</li> <li>Dogs</li>
      </ul>
  </body> </html>
```

आम तौर पर, जब आप अपने वेब ब्राउज़र का उपयोग कर रहे होते हैं, तो आप उन पृष्ठों को देख रहे होते हैं जिन्हें इंटरनेट पर होस्ट किया जाता है। हालांकि, ब्राउज़र HTML दस्तावेजों को प्रदर्शित करने में पूरी तरह से अच्छे हैं जो आपके स्थानीय मशीन पर भी हैं। आपको दिखाने के लिए कि मेरा क्या मतलब है, मैं आपको एक पाठ संपादक को खोलने और उदाहरण में कोड दर्ज करने के लिए आमंत्रित करता हूं।

Picking the Right Text Editor

कुछ Text editors HTML को संलेखन के लिए अनुकूल नहीं हैं। विशेष रूप से, आप बचना चाहते हैं editors जो रिच टेक्स्ट एडिटिंग का समर्थन करते हैं, जैसे Microsoft WordPad (Windows) या TextEdit (मैक ओएस एक्स)। इस प्रकार के editors plain text के अलावा अन्य स्वरूपों में अपनी फ़ाइलों को सहेज सकते हैं, जिससे आपका HTML टूट जाएगा। यदि आपको TextEdit का उपयोग करना है, तो चयन करके plain text में सहेजें प्रारूप → प्लेन टेक्स्ट बनाएं। विंडोज में, वर्डपैड के बजाय नोटपैड का उपयोग करें।

यदि आप एक अच्छे टेक्स्ट एडिटर के लिए मार्केट में हैं, तो मैक पर मेरी सिफारिश है TextMate। विंडोज के लिए ई टेक्स्ट एडिटर नामक एक क्लोन संस्करण है। यदि मुफ्त आपकी चीज़ है, तो आप

मैक के लिए टेक्स्ट रैंगलर डाउनलोड कर सकते हैं। विंडोज के लिए, नोटपैड 2 और नोटपैड ++ को अत्यधिक माना जाता है। लिनक्स पाठ के वर्गीकरण के साथ आता है
संपादकों, जैसे vi, नैनो, emacs और gedit।

जब आप उदाहरण 1–3 से कोड दर्ज कर रहे हों, तो इसे अपने डेस्कटॉप पर सहेजें a.html और फिर क्रोम एपलिकेशन आइकन पर फ़ाइल को खींचकर या क्रोम को खोलकर या फ़ाइल का चयन करके क्रोम के साथ इसे खोलें?खुली फाइल। डबल क्लिक a.html भी काम करेगा, लेकिन यह आपकी सेटिंग के आधार पर आपके टेक्स्ट एडिटर या किसी अन्य ब्राउज़र में खुल सकता है।

यहां तक कि अगर आप मैक ओएस एक्स नहीं चला रहे हैं, तो आपको डेस्कटॉप वेब ब्राउज़र पर अपने एंड्रॉइड वेब ऐप का परीक्षण करते समय क्रोम का उपयोग करना चाहिए, क्योंकि क्रोम एंड्रॉइड के मोबाइल ब्राउज़र के लिए निकटतम डेस्कटॉप ब्राउज़र है।

Introduction to CSS

जैसा कि आपने देखा है, ब्राउज़र कुछ HTML elementों को अलग–अलग शैलियों के साथ प्रस्तुत करते हैं (उदाहरण के लिए, शीर्ष लेख बड़े और बोल्ड हैं, पैराग्राफ एक रिक्त पंक्ति और उसके बाद आगे आते हैं)। ये शैली बहुत बुनियादी हैं और मुख्य रूप से पाठक को दस्तावेज़ की संरचना और अर्थ के तहत खड़े होने में मदद करने के लिए अभिप्रेत हैं।

इस सरल संरचना–आधारित प्रतिपादन से परे जाने के लिए, आप कैस्केडिंग स्टाइल शीट्स (सीएसएस) का उपयोग करते हैं। CSS एक स्टाइलशीट भाषा है जिसका उपयोग आप HTML डॉक्यूमेंट के विजुअल प्रेजेंटेशन को परिभाषित करने के लिए करते हैं। आप

सीएसएस का उपयोग टेक्स्ट रंग, आकार और शैली (बोल्ड, इटैलिक, आदि) जैसी सरल चीजों को परिभाषित करने के लिए कर सकते हैं, या पेज लेआउट, ग्रेडिएंट, अपारदर्शिता, और बहुत कुछ जैसी जटिल चीजें।

उदाहरण 1–4 एक सीएसएस नियम दिखाता है जो ब्राउज़र को निर्देश देता है कि वह रंग लाल का उपयोग करते हुए body के किसी भी element को प्रदर्शित करे। इस उदाहरण में, body चयनकर्ता है (यह निर्दिष्ट करता है कि नियम से क्या प्रभावित होता है) और घुंघराले ब्रेसिज़ घोषणा (नियम खुद) को घेरते हैं। घोषणा में संपतियों और उनके मूल्यों का एक सेट शामिल है। इस उदाहरण में, रंग संपति है, और लाल रंग संपति का मूल्य है।

Example 1-4. A simple CSS rule

```
body { color: red; }
```

Property के नाम पूर्वनिर्धारित CSS विनिर्देशन में पूर्वनिर्धारित हैं, जिसका अर्थ है कि आप बस उन्हें नहीं बना सकते हैं। प्रत्येक Property एक उचित मूल्य की उम्मीद करती है, और किसी दिए गए Property के लिए बहुत सारे उचित मूल्य और मूल्य प्रारूप हो सकते हैं।

उदाहरण के लिए आप रंगों को लाल या एचटीएमएल रंग कोड नोटेशन का उपयोग करके निर्दिष्ट कर सकते हैं जो हेक्साडेसिमल नोटेशन का उपयोग करता है: एक हैश / पाउंड साइन (#) जिसके बाद हेक्साडेसिमल अंकों के तीन जोड़े (0–F) बाएं से दाएं लाल का प्रतिनिधित्व करते हैं हरे और नीले मान (red is represented as #FF0000). गुण जो खसरे की अपेक्षा करते हैं, वे 10px, 75% और 1em जैसे मूल्यों को स्वीकार कर सकते हैं। उदाहरण 1–5 कुछ

सामान्य घोषणाओं को दर्शाता है। पृष्ठभूमि और रंग के लिए दिखाया गया रंग कोड CSS "ग्रे–" से मेल खाता है

Example 1-5. Some common CSS declarations

```
body {
   color: red;
   background-color: #808080;     font-size: 12px;
   font-style: italic;
   font-weight: bold;
   font-family: Arial;
}
```

चयनकर्ता विभिन्न प्रकार के स्वादों में आते हैं। यदि आप अपने सभी हाइपरलिंक (एक element) को इटैलिक में प्रदर्शित करना चाहते हैं, तो अपनी स्टाइलशीट में निम्न जोड़ें:

```
a { font-style: italic; } }
```

यदि आप अधिक विशिष्ट होना चाहते हैं और केवल उस हाइपरलिंक को इटैलिकाइज करते हैं जो h1 टैग में कहीं निहित है, तो अपनी स्टाइलशीट में निम्नलिखित जोड़ें।:

```
h1 a {font–style: italic; }
```

आप अपने HTML टैग्स में आईडी और / या class विशेषताओं को जोड़कर अपने स्वयं के कस्टम चयनकर्ताओं को भी परिभाषित कर सकते हैं। निम्नलिखित HTML स्निपेट पर विचार करें:

```
<h1 class="loud">Hi there!</h1>
<p>Thanks for visiting my web page.</p> <p>I hope you like
it.</p>
<ul>
    <li class="loud">Pizza</li>
        <li>Beer</li>
```

```
    <li>Dogs</li>
</ul>
```

अगर हम जोड़ते हैं। add .loud { font-style: italic; } इस HTML के लिए सीएसएस के लिए, हाय! और पिज्जा को इटैलिकाइज़ किया जाएगा, क्योंकि उन दोनों की ज़ोर क्लास है। .loud सेलेक्टर महत्वपूर्ण है – यह है कि कैसे सीएसएस जोर से एक class के साथ HTML टैग देखने के लिए जानता है। यदि आप डॉट को छोड़ देते हैं, तो सीएसएस एक ज़ोर से टैग की तलाश करेगा, जो इस स्निपेट में मौजूद नहीं है (या HTML में उस मामले के लिए)।

सीएसएस को आईडी द्वारा लागू करना समान है। हाइलाइट पैरा–ग्राफ टैग में एक पीले रंग की पृष्ठभूमि को भरने के लिए, निम्नलिखित नियम का उपयोग करें:

```
#highlight { background-color: yellow; }
```

यहां # चिह्न सीएसएस को आईडी हाइलाइट के साथ HTML टैग देखने के लिए कहता है।

पुनर्कथन करने के लिए, आप कक्षा नाम (जैसे, .loud, .subtle, . eelementr), या ID (जैसे, #highlight, #login, #promo) द्वारा टैग नाम (जैसे, body, h1, p) के elements का चयन कर सकते हैं।)। और, आप चयनकर्ताओं को एक साथ चुनकर और अधिक विशिष्ट प्राप्त कर सकते हैं (जैसे, h1 a, body ul, .loud)।

CLASS और ID के बीच अंतर हैं। जब आप एक ही class मान वाले पृष्ठ पर एक से अधिक आइटम रखते हैं, तो class विशेषताओं का उपयोग करें। इसके विपरीत, आईडी मान एक पृष्ठ के लिए UNIQUE होना चाहिए।

जब मैंने पहली बार यह सीखा, तो मुझे लगा कि मैं हमेशा क्लास की विशेषताओं का उपयोग करूंगा, इसलिए मुझे इस बारे में चिंता करने की ज़रूरत नहीं है कि क्या मैं एक आईडी मूल्य को धोखा दे रहा था। हालाँकि, ID द्वारा element का चयन करना class की तुलना में बहुत तेज़ है, इसलिए आप class चयनकर्ताओं को अत्यधिक प्रभावित करके अपने प्रदर्शन को प्रभावित कर सकते हैं।

Applying a stylesheet

तो अब आप CSS की मूल बातें समझ गए हैं, लेकिन आप HTML पेज पर स्टाइलशीट कैसे लागू करते हैं? वास्तव में सरल, वास्तव में! सबसे पहले, आप अपने सर्वर पर सीएसएस को कहीं भी सहेजते हैं (आमतौर पर आपकी HTMM फ़ाइल के समान निर्देशिका में, हालांकि आप इसे subdirectory में डाल सकते हैं)। अगला, HTML दस्तावेज़ के head में स्टाइलशीट के लिए लिंक, जैसा कि उदाहरण में दिखाया गया है. इस उदाहरण में href विशेषता एक सापेक्ष पथ है, जिसका अर्थ है कि यह screen.css नामक एक पाठ फ़ाइल की ओर इशारा करता है। HTML पेज के समान डायरेक्टरी में आप निरपेक्ष लिंक भी निर्दिष्ट कर सकते हैं, जैसे कि निम्नलिखित:

http://example.com/screen.css

यदि आप अपनी HTML फ़ाइलों को अपनी स्थानीय मशीन पर सहेज रहे हैं, तो आप चीजों को सरल रखना चाहते हैं: CSS फ़ाइल को HTML फ़ाइल में उसी निर्देशिका में रखें और उदाहरण के रूप में दिखाए गए अनुसार एक सापेक्ष पथ का उपयोग करें.

Example 1-6. Linking to a CSS stylesheet

```
<html> <head> <title>My Awesome Page</title>
<link rel="stylesheet" href="screen.css" type="text/css" />
 </head>
 <body>
<h1 class="loud">Hi there!</h1>
```

```
<p>Thanks for visiting my web page.</p>
<p>I hope you like it.</p>
<ul>

<li class="loud">Pizza</li>
<li>Beer</li>
<li>Dogs</li>

</ul> </body> </html>
```

उदाहरण 1–7 screen.css की सामग्री को दर्शाता है। आपको इस फ़ाइल को HTML फ़ाइल के समान स्थान पर सहेजना चाहिए:

Example 1-7. A simple stylesheet

```
body {
font-size: 12px;
font-weight: bold;
font-family: Arial;
}
a { font-style: italic; }
h1 a { font-style: italic; }

.loud { font-style: italic; }
#highlight { background-color: yellow; }
```

यह इंगित करने योग्य है कि आप होस्ट किए गए स्टाइलशीट से लिंक कर सकते हैं HTML दस्तावेज़ को होस्ट करने के अलावा अन्य डोमेन पर। तथापि, बिना किसी और के स्टाइलशीट से लिंक करना बहुत अशिष्ट माना जाता है अनुमति है, इसलिए कृपया केवल अपने आप से लिंक करें।

Introduction to JavaScript

इस बिंदु पर आप जानते हैं कि HTML के साथ दस्तावेज़ को कैसे संरचित करना है और सीएसएस के साथ इसकी दृश्य प्रस्तुति को कैसे संशोधित किया जाए। अब हम कुछ जावास्क्रिप्ट जोड़ेंगे ताकि

यह सामान बना सके।

जावासक्रिप्ट एक सक्रिपटिंग भाषा है जिसे आप उपयोगकर्ता के लिए अधिक इंटरैक्टिव और सुविधाजनक बनाने के लिए एक HTML पेज में जोड़ सकते हैं। उदाहरण के लिए, आप कुछ जावासक्रिप्ट लिख सकते हैं जो टाइप किए गए मूल्यों का निरीक्षण करेंगे ताकि यह सुनिश्चित हो सके कि वे मान्य हैं। या, जहां उपयोगकर्ता क्लिक करता है, उसके आधार पर आपके पास जावासक्रिप्ट शो हो सकता है या पृष्ठ के element को छिपा सकता है। जावासक्रिप्ट वर्तमान वेब पेज को फिर से ताज़ा किए बिना डेटाबेस परिवर्तनों को निष्पादित करने के लिए वेब सर्वर से भी संपर्क कर सकता है।

किसी भी आधुनिक सक्रिपटिंग भाषा की तरह, जावासक्रिप्ट में चर, सरणियाँ, ऑब्जेक्ट और सभी विशिष्ट नियंत्रण संरचनाएं हैं (जैसे, यदि, जबकि, के लिए)। उदाहरण 1–8 जावासक्रिप्ट का एक स्निपेट दिखाता है जो भाषा की कई मुख्य अवधारणाओं को दिखाता है।

उदाहरण 1–8. बेसिक जावासक्रिप्ट सिंटैक्स

```
var foods = ['Apples',  'Bananas',  '0ianges'],(
for (var i = O; i <foods.length( i++ {
if (foods[i] == 'Apples'{
alert(foods[i]+' are my favourite!');
}else{
alert(foods[i]+' are okay');
}
}}
```

यहाँ क्या हो रहा है इसका स्पष्टीकरण दिया गया है:

1. तीन element वाले खाद्य पदार्थों की एक सरणी (मूल्यों की

सूची) को परिभाषित करें।

2. लूप के लिए एक विशिष्ट खोलें जो i से 0 नाम के एक चर को इनिशियलाइज़ करता है और इस मामले में एक एग्जिट मानदंड को निर्दिष्ट करता है, बाहर निकलें जब मैं खाद्य पदार्थ सरणी की लंबाई से अधिक हो जाता है, और लूप के माध्यम से 1 बार बढ़ाता हूं (i ++ के लिए शॉर्टहैंड है) "I के वर्तमान मूल्य में 1 जोड़ें")
3. एक बगीचे की विविधताए जो यह देखने के लिए जांचती है कि क्या सरणी का वर्तमान element बराबर है।
4. प्रदर्शित किया जाता है यदि सरणी का वर्तमान element सेब के बराबर है।
5. प्रदर्शित किया जाता है यदि सरणी का वर्तमान element सेब के बराबर नहीं है।

यहां जावासक्रिप्ट के सिंटैक्स के बारे में कुछ बिंदु दिए गए हैं जो ध्यान देने योग्य हैं:

- Statements अर्धविराम के साथ समाप्त किए जाते हैं (;)
- कोड ब्लॉक ब्रेसिज़ में संलग्न हैं ({})
- वेरिएबल्स को var कीवर्ड का उपयोग करके घोषित किया जाता है
- ऐरे elements को class ब्रैकेट संकेतन के साथ पहुँचा जा सकता है ([])
- ऐरे कीज़ को 0 पर शुरू किया गया है
- एकल बराबर चिह्न (=) असाइनमेंट ऑपरेटर है (किसी वैरिएबल के लिए मान प्रदान करता है))
- डबल बराबर चिह्न (==) समतुल्य तार्किक संचालक है (दो मूल्यों की तुलना करता है और यदि वे समकक्ष हैं तो सत्य

का मूल्यांकन करते हैं)

- प्लस चिन्ह (+) स्ट्रिंग समवर्ती ऑपरेटर है (दो तारों को जोड़ता है

हमारे उद्देश्यों के लिए, जावास्क्रिप्ट की सबसे महत्वपूर्ण विशेषता यह है कि इसके साथ बातचीत कर सकते हैं एचटीएमएल पेज के element

उदाहरण 1–9 जावास्क्रिप्ट का एक सरल सा दिखाता है जो पृष्ठ पर कुछ पाठ को बदलता है उपयोगकर्ता h1 पर क्लिक करता है।

DOM का उद्देश्य डॉक्यूमेंट ऑब्जेक्ट मॉडल है और इस संदर्भ में यह HTML पेज की browser की समझ को पुन: प्रस्तुत करता है। आप यहां DOM के बारे में अधिक जानकारी प्राप्त कर सकते हैं: http://en.wikipedia.or8/wiki/Documer1t_Object_ModeL

Example 1-9. Simple onclick handler

```
<html>
    <head>
        <title>My Awesome Page</title>
        <script type="text/javascript" charset="utf-8"> ❶
            function sayHello() { ❷
                document.getElementById('foo').innerHTML = 'Hi there!'; ❸
            } ❹
        </script> ❺
    </head>
    <body>
        <h1 id="foo" onclick❻="sayHello()">Click me!</h1>
    </body>
</html>
```

यहाँ एक स्पष्टीकरण है:

- HTML दस्तावेज़ के Head पर एक सक्रिप्ट ब्लॉक।
- यह लाइन सक्रिप्ट ब्लॉक के अंदर एक एकल जावास्क्रिप्ट फ़ंक्शन कहती है जिसका नाम sayHello() है।
- SayHello () फ़ंक्शन में एक एकल स्टेटमेंट होता है, जो ब्राउज़र को उस element के लिए दस्तावेज़ के माध्यम से

देखने के लिए कहता है, जिसमें आईडी फू है, और अपने आंतरिक HTML सामग्री को हाय में सेट करें! ब्राउज़र में इसका प्रभाव यह है कि पाठ "मुझे क्लिक करें!" "हाय वहाँ!" जब उपयोगकर्ता h1 element पर क्लिक करता है।

- अंत हेलो () फ़ंक्शन।
- सक्रिप्ट ब्लॉक का अंत।
- () HI element की ऑनक्लिक विशेषता ब्राउज़र को कुछ करने के लिए कहती है, जब उपयोगकर्ता h1 element पर क्लिक करता है, अर्थात्, sayHello () फ़ंक्शन को चलाने के लिए।

वेब विकास के बुरे पुराने दिनों में, विभिन्न ब्राउज़रों का जावास्क्रिप्ट के लिए अलग समर्थन था। इसका मतलब है कि आपका कोड सफारी 2 में चल सकता है लेकिन इंटरनेट एक्सप्लोरर में नहीं6. आपको यह सुनिश्चित करने के लिए कि आपके कोड सभी के लिए काम करेंगे, प्रत्येक ब्राउज़र (और एक ही ब्राउज़र के अलग–अलग version) का परीक्षण करने के लिए आपको दर्द उठाना पड़ा। जैसे–जैसे ब्राउज़र और ब्राउज़र version की संख्या बढ़ती गई, वैसे–वैसे हर परिवेश के लिए आपके जावास्क्रिप्ट कोड का परीक्षण और रखरखाव असंभव हो गया। उस समय ,JavaScript के साथ वेब प्रोग्रामिंग नरक थी।

JQuery दर्ज करें। jQuery एक अपेक्षाकृत छोटा जावास्क्रिप्ट पुस्तकालय है जो आपको अपने जावास्क्रिप्ट कोड को इस तरह से लिखने की अनुमति देता है जो विस्तृत ब्राउज़रों में समान काम करेगा। क्या अधिक है, यह कई सामान्य वेब विकास कार्यों को सरल करता है। इन कारणों से, मैं अपने अधिकांश वेब विकास कार्यों में jQuery का उपयोग करता हूं, और मैं इस पुस्तक में जावास्क्रिप्ट उदाहरणों के लिए इसका उपयोग करूंगा। उदाहरण 1–10

Example 1-10. jQuery onclick handler

```
<html>
    <head>
        <title>My Awesome Page</title>
        <script type="text/javascript" src="jquery.js"></script> ❶
        <script type="text/javascript" charset="utf-8">
            function sayHello() {
                $('#foo').text('Hi there!'); ❷
            }
        </script>
    </head>
    <body>
        <h1 id="foo" onclick="sayHello()">Click me!</h1>
    </body>
</html>
```

- इस लाइन में द एज्क्यूरी शामिल है।जेएस पुस्तकालय। यह एक सापेक्ष पथ है, जिसका अर्थ है कि फ़ाइल उसी निर्देशिका में मौजूद है जिस पृष्ठ का उपयोग किया जा रहा है (यह उदाहरण जीआरआई लाइब्रेरी, jquer y तक लगभग कार्य नहीं करेगा।जे एस, वहाँ है)। हालाँकि, आप इसे विभिन्न स्थानों से सीधे शामिल कर सकते हैं जहाँ यह उपलब्ध है।
- एचएल element में पाठ को बदलने के लिए हमें कोड की मात्रा में कमी की सूचना देनी होगी। इस तरह के तुच्छ उदाहरण में यह एक बड़ी बात नहीं हो सकती है, लेकिन मैं आपको आश्वस्त कर सकता हूं कि यह जटिल समाधानों में एक जीवनरक्षक है।

हम वास्तविक दुनिया के बहुत सारे jQuery के उदाहरणों को बाद में देख रहे हैं, इसलिए मैं इसे उसी समय छोड़ने जा रहा हूं।

http://jquerjy.com पर jQuery के डाउनलोड, डॉक्यूमेंसियन और कटारियल उपलब्ध हैं : उदाहरण के लिए 1–9 में दिखाए गए jQuery का उपयोग करने के लिए, आपको इसे वहां से डाउनलोड करना होगा, आपके द्वारा डाउनलोड किए गए फ़ाइल का नाम बदलना होगा (जैसे कि – jquery-1.4.2.min.js) से jquery.js और आपके

HTML डॉक्यूमेंट के समान समान directory में copy की एक प्रति चिपकाएँ।

CHAPTER 2

BASIC STYLING

अंततः, हम HTML, CSS और JavaScript का उपयोग करके एक native Android ऐप बनाने जा रहे हैं। इस यात्रा पर पहला कदम आरामदायक स्टाइलिंग एचटीएमएल को देखने के लिए मिलता है मोबाइल एप्लिकेशन।

इस अध्याय में, मैं आपको दिखाता हूँ कि CSS styles को मौजूदा HTML पृष्ठों के एक समूह में कैसे लागू किया जाए ताकि वे एंड्रॉइड फोन पर आसानी से नेविगेट कर सकें। तो, के अलावा एक मूल एप्लिकेशन के निर्माण के करीब जा रहे हैं, आप एक व्यावहारिक (और मूल्यवान) कौशल सीख रहे होंगे जिसे आप तुरंत उपयोग कर सकते हैं।

Don't Have a Website?

यदि आप अपने व्यक्तिगत कंप्यूटर पर अपने सभी वेब पेजों का परीक्षण कर रहे हैं, तो आप सर्वर स्थापित किए बिना उन्हें अपने एंड्रॉइड फोन पर नहीं देख पाएंगे। तुम्हारे पास एक कई विकल्प हैं:

- अपने वेब पेज को एक वेब सर्वर पर होस्ट करें और अपने एंड्रॉइड फोन से उस सर्वर से कनेक्ट करें। संभावनाएं अच्छी हैं कि आपका इंटरनेट सेवा प्रदाता (आईएसपी) मानार्थ वेब होस्टिंग प्रदान करता है, लेकिन यह आमतौर पर HTML जैसी बुनियादी सुविधाओं का समर्थन करता है। जब तक हम अध्याय 6 तक पहुँचते हैं, तब तक हमें PHP का उपयोग करने की आवश्यकता होती है, एक सक्रिप्टिंग भाषा जो वेब सर्वर पर चलती है, इसलिए आपको एक सस्ती होस्टिंग पर ध्यान देना चाहिए। कई कंपनियां, जैसे लाफिंग स्क्वीड, के साथ एंट्री–लेवल होस्टिंग प्रदान करती हैं $10 एक महीने के लिए PHP।

- अपने कंप्यूटर पर चलने वाले वेब सर्वर पर उन्हें होस्ट करें और अपने एंड्रॉइड फोन से अपने कंप्यूटर पर चल रहे वेब सर्वर से कनेक्ट करें। यह तभी काम करता है जब आपका Android फ़ोन और कंप्यूटर एक ही WiFi नेटवर्क पर हैं।

यह अध्याय सेट किया गया है ताकि आप उदाहरणों की कोशिश कर सकें जैसे कि आप इसके माध्यम से जाते हैं। इसलिए, कोई फर्क नहीं पड़ता कि आप वेब पेज देखने के लिए कौन सा विकल्प चुनते हैं, उन्हें ब्राउज़र में अधिमानतः लोड करने का प्रयास करें (अधिमानतः एंड्रॉइड ब्राउज़र) जब भी आप नमूनों में से एक में कुछ नया जोड़ते हैं। हालाँकि, अपने फ़ाइल संपादक में अपनी फ़ाइल सहेजने से पहले सुनिश्चित करें कि आप इसे पुनः लोड करें ब्राउज़र या आप अपने परिवर्तन नहीं देखेंगे।

Running a Web Server Locally

सभी मौजूदा मुख्यधारा ऑपरेटिंग सिस्टम (लिनक्स, विंडोज, मैक ओएस एक्स) में कुछ प्रकार के वेब सर्वर शामिल हैं। Mac OS X पर, सिस्टम प्राथमिकताएँ खोलें, साझाकरण चुनें और सक्षम करें वेब शेयरिंग। वेब शेयरिंग शुरू करने के बाद, वेब शेयरिंग प्राथमिकताएं आपकी व्यक्तिगत वेबसाइट का URL प्रदर्शित करेंगी (इसमें आपके द्वारा डाली गई कोई भी चीज शामिल है साइट निर्देशिका आपके home directory में)। यह फ़ॉर्म http://local-hostname/ ~your-username. आपके उपयोगकर्ता नाम का होगा।

लिनक्स के कुछ संस्करणों पर, जैसे कि उबंटू, आपको वेब सर्वर को स्थापित करने और सक्षम करने के लिए कुछ अतिरिक्त चरणों से गुजरना होगा। सबसे पहले, Apache को कमांड पर इंस्टॉल करें sudo एप्टीट्यूड के साथ लाइन Apache2 स्थापित करें। इसके बाद, उपयोगकर्ता निर्देशिका मॉड्यूल को सक्षम करें sudo a2enmod

userdir। एक बार ऐसा करने के बाद, Apache को इस कमांड के साथ पुनः आरंभ करें: sudo /etc/init.d/apache2 restart करें। आपके द्वारा ऐसा करने के बाद, आप एक निर्देशिका बना सकते हैं
आपके होम डायरेक्टरी में public_html कहा जाता है और किसी भी फाइल को ऐसे URL के साथ एक्सेस करते हैं
http://local-hostname/~your-username के रूप में।

विंडोज पर, चीजें थोड़ी अधिक शामिल हैं। आपको एक संस्करण चलाने की आवश्यकता है विंडोज़ जिसमें इंटरनेट इनफॉर्मेशन सर्विसेज (IIS) शामिल हैं और इसमें जाकर सक्षम करें नियंत्रण कक्ष और विंडोज सुविधाओं को चालू या बंद करने के लिए अपना रास्ता बना रहा है। उपरांत आपने ऐसा किया है, अपने वेब दस्तावेज़ों को अपने IIS दस्तावेज़ रूट में डालें, जो कि आमतौर पर है C:\inetpub\wwwroo पर स्थित है। यदि आप विंडोज पर अपाचे को चलाना पसंद करते हैं, तो देखें इस तरह के EasyPHP के रूप में एक prepackaged समाधान, या इस विषय पर विकिपीडिया पृष्ठ की जाँच करें
http://en.wikipedia.org/wiki/Comparison_of_WAMPs.

First Steps

कल्पना कीजिए कि आपके पास एक वेबसाइट है जिसे आप "2–1 जुटाना" चाहते हैं. इस परिदृश्य में, एंड्रॉइड के लिए साइट को अनुकूलित करने के लिए कई आसान चीजें हैं जो आप कर सकते हैं। मैं इस अध्याय में आपके विकल्पों पर जाऊंगा।चित्र 2–2 दिखाता है कि यह वेब पेज एंड्रॉइड फोन पर कैसा दिखता है। यह प्रयोग करने योग्य है, लेकिन Android के लिए अनुकूलित से बहुत दूर है।

उदाहरण 2–1 चित्र 2–2 में दिखाया गया वेबसाइट का संक्षिप्त संस्करण दिखाता है. यह वह HTML है जिसके साथ आप इस

अध्याय में काम करेंगे। आप इसे इस पुस्तक से डाउनलोड कर सकते हैं। यदि आप अध्याय के माध्यम से जाना चाहते हैं तो वेबसाइट इसे स्टाइल करना चाहेगी। डेस्कटॉप Stylesheet.css (स्क्रीन) नहीं दिखाया गया है क्योंकि यह आवश्यक नहीं है, लेकिन आप इससे स्टाइलशीट का उपयोग कर सकते हैं। पिछले अध्याय यदि आप के साथ खेलने के लिए कुछ करना चाहते हैं।

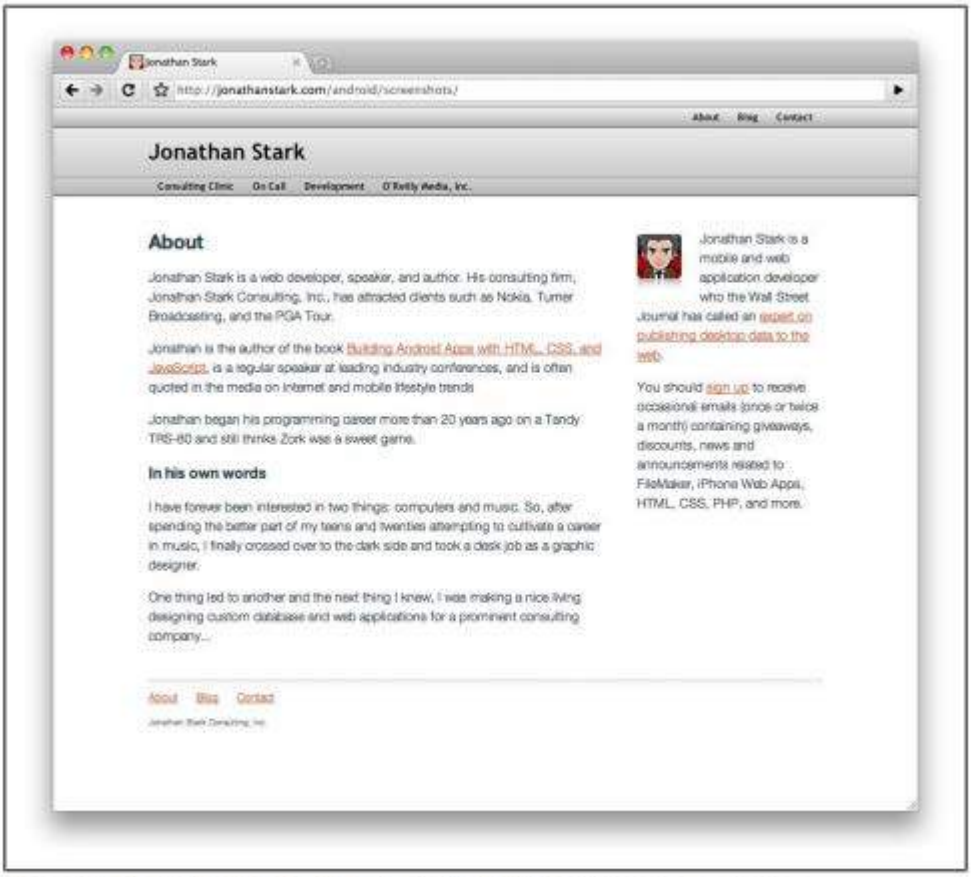


Figure 2-1. Desktop version of a typical website looks fine in Chrome on a desktop

Example 2-1. The HTML document we'll be styling

```
<html>
<head>
  <link rel="stylesheet" href="screen.css" type="text/css" />
  <title>Jonathan Stark</title>
</head>
<body>
<div id="container">
  <div id="header">
    <h1><a href="./">Jonathan Stark</a></h1>
    <div id="utility">
        <ul>
            <li><a href="about.html">About</a></li>
            <li><a href="blog.html">Blog</a></li>
            <li><a href="contact.html">Contact</a></li>
        </ul>
    </div>
    <div id="nav">


         <ul>
             <li><a href="consulting-clinic.html">Consulting Clinic</a></li>
             <li><a href="on-call.html">On Call</a></li>
             <li><a href="development.html">Development</a></li>
             <li><a href="http://www.oreilly.com">O'Reilly Media, Inc.</a></li>
         </ul>
    </div>
  </div>
  <div id="content">
    <h2>About</h2>
    <p>Jonathan Stark is a web developer, speaker, and author. His
       consulting firm, Jonathan Stark Consulting, Inc., has attracted
       clients such as Staples, Turner Broadcasting, and the PGA Tour.
       ...
       </p>
  </div>
  <div id="sidebar">
    <img alt="Manga Portrait of Jonathan Stark"
         src="jonathanstark-manga-small.png"/>
    <p>Jonathan Stark is a mobile and web application developer who the
       Wall Street Journal has called an expert on publishing desktop
       data to the web.</p>
  </div>
  <div id="footer">
    <ul>
        <li><a href="services.html">Services</a></li>
        <li><a href="about.html">About</a></li>
        <li><a href="blog.html">Blog</a></li>
    </ul>
    <p class="subtle">Jonathan Stark Consulting, Inc.</p>
  </div>
</div>
</body>
</html>
```

सालों तक, वेब डेवलपर्स एक ग्र डि में elements को बाहर करने के लिए तालिकाओं का उपयोग करते थे। CSS और HTML में अग्र मिों ने न केवल अप्रचलित, बल्कि अवांछनीय दृष्टिकोण का प्रतिपादन किया है। आज, हम मुख्य रूप से div element (साथ में) का उपयोग करते हैं एक ही चीज़ को पूरा करने के लिए कई प्रकार की

विशेषताओं के साथ), लेकिन अधिक के साथ नियंत्रण। हालांकि डिव–आधारित लेआउट की पूरी व्याख्या अच्छी तरह से है इस पुस्तक के दायरे के बाहर, आपको इसके बहुत सारे उदाहरण दिखाई देंगे पुस्तक के माध्यम से पढ़ें। अधिक जानने के लिए, कृपया डिजाइनिंग की जाँच करें जेफ़री ज़ेल्डमैन (न्यू राइडर प्रेस) द्वारा वेब मानक, जो कवर करता है अधिक विस्तार से जारी करें।

Figure 2-2. Desktop version of a typical website looks all right on an Android phone, but we can do a lot better

Prepare a Separate Android Stylesheet

मैं अगले आदमी के रूप में DRY हूं, लेकिन असली दुनिया में आप एक बेहतर ब्रेक बनाने से बेहतर हैं आपके डेस्कटॉप ब्राउज़र स्टाइलशीट और आपके एंड्रॉइड स्टाइलशीट के बीच। मेरा वचन लो इसके लिए और बस दो पूरी तरह से स्वतंत्र फाइलें बनाएं; आप बेहतर नींद लेंगे। विकल्प यह है कि आप अपने सभी सीएसएस नियमों को एक एकल स्टाइलशीट में बदल दें, जो कि एक बुरा विचार है कारणों की संख्या, जिनमें से सबसे स्पष्ट है कि आप का एक गुच्छा भेज देंगे फोन के लिए अप्रासंगिक डेस्कटॉप शैली नियम है, जो कीमती बैंडविड्थ और मेमोरी की बर्बादी है।

DRY का अर्थ है अपने आप को दोहराना नहीं है, और यह एक

सॉफ्टवेयर डेवलपमेंट सिद्धांत है जो बताता है, ज्ञान के हर टुकड़े में एक प्रणाली के भीतर एक एकल, अस्पष्ट, आधिकारिक प्रतिनिधित्व होना चाहिए। अवधि एंड्रयू हंट और डेविड थॉमस ने अपनी पुस्तक द प्रैजमैटिक प्रोग्रामर (एडिसन–वेस्ले प्रोफेशनल) में गढ़ा था।

विशेष रूप से एंड्रॉइड के लिए एक स्टाइलशीट निर्दिष्ट करने के लिए, निम्नलिखित HTML का उपयोग करने वाले लोगों के साथ नमूना HTML दस्तावेज़ में स्टाइलशीट लिंक टैग को बदलें:

```
<link rel="stylesheet" type="text/css"
 href="android.css" media="only screen and (max-width: 480px)"
/>
<link rel="stylesheet" type="text/css"
 href="desktop.css" media="screen and (min-width: 481px)" />
```

मैंने विशेष रूप से यहां अधिकतम–चौड़ाई और न्यूनतम–चौड़ाई का उपयोग किया है ताकि आप अपने डेस्कटॉप ब्राउज़र का आकार बदल सकें और पृष्ठ का मोबाइल संस्करण देख सकें। यदि आप डेस्कटॉप परोसना पसंद करते हैं। अपने ब्राउज़र विंडो के आकार की परवाह किए बिना डेस्कटॉप उपयोगकर्ताओं के लिए सीएसएस स्टाइलशीट, इसके बजाय अधिकतम डिवाइस–चौड़ाई और माइंडवाइस–चौड़ाई का उपयोग करें। वायरलेस यूनिवर्सल रिसोर्स फ़ाइल (WURFL) में जानकारी होती है।

आप Android उपकरणों सहित भारी संख्या में वायरलेस उपकरणों की पहचान करने के लिए उपयोग कर सकते हैं। अगर आपको 480px (जैसे कि टैबलेट) से अधिक की चौड़ाई वाले एंड्रॉइड डिवाइस का पता लगाना है या यदि आप मोबाइल नहीं चाहते हैं साइट का संस्करण जब उपयोगकर्ता अपनी ब्राउज़र विंडो को 480px से कम आकार में प्रदर्शित करता है, तो आप विशिष्ट ब्राउज़रों का

सटीक पता लगाने के लिए WURFL के PHP API का उपयोग कर सकते हैं। WURFL के बारे में अधिक जानकारी के लिए परिशिष्ट देखें।

यहाँ, desktop.css आपके मौजूदा डेस्कटॉप स्टाइलशीट और एंड्रॉइड को संदर्भित करता है। एक नई फाइल है जिस पर हम विस्तार से चर्चा करेंगे। desktop.css फ़ाइल आवश्यक नहीं है, लेकिन आप यदि आप चाहें तो पिछले अध्याय के स्टाइलशीट का उपयोग कर सकते हैं।

यदि आप उदाहरण 2–1 में दिखाए गए नमूना HTML दस्तावेज़ का उपयोग कर रहे हैं, तो आपको स्क्रीन का नाम बदलना होगा।डेस्कटॉप के लिए सीएसएस।सीएसएस, लेकिन जब से हम एंड्रॉइड स्टाइलशीट पर ध्यान केंद्रित कर रहे हैं, आप डेस्कटॉप स्टाइलशीट को पूरी तरह से अनदेखा कर सकते हैं। यदि यह लोड करने में विफल रहता है, तो आपका ब्राउज़र भी नहीं मिलेगा परेशान।

हालाँकि, यदि आप साइट के Android–अनुकूलित संस्करण का परीक्षण करने के लिए Chrome का उपयोग करना चाहते हैं, तो आपको डेस्कटॉप के संदर्भ को प्रतिस्थापित करना चाहिए।के साथ सीएसएस Android के लिए एक संदर्भ।सीएसएस। इस तरह, आपको अपनी साइट का Android संस्करण चलाना पड़ेगा, चाहे आप इसे फ़ोन या डेस्कटॉप ब्राउज़र से लोड करें।

अफसोस, इंटरनेट एक्सप्लोरर इन अभिव्यक्तियों को नहीं समझेगा, इसलिए हमें एक सशर्त टिप्पणी (बोल्ड में दिखाया गया है) जोड़ना होगा जो सीएसएस के डेस्कटॉप संस्करण के लिए लिंक करता है:

```
<link rel="stylesheet" type="text/css"
 href="android.css" media="only screen and (max-width: 480px)"
/>
<link rel="stylesheet" type="text/css"
 href="desktop.css" media="screen and (min-width: 481px)" />
<!--[if IE]>
<link rel="stylesheet" type="text/css" href="explorer.css"
media="all" />
<![endif]-->
```

तो अब HTML डॉक्युमेंट को एडिट करने का समय आ गया है (यदि आपने पहले से ऐसा नहीं किया है जैसा कि आप साथ कर रहे थे): स्क्रीन के मौजूदा लिंक को हटा दें।सीएसएस फ़ाइल, और इसे के साथ बदलें लाइनों बस दिखाया। इस तरह, आपके पास इस अध्याय में एंड्रॉइड–विशिष्ट सीएसएस के लिए एक साफ स्लेट होगा।

Control the Page Scaling

जब तक आप इसे अन्यथा नहीं बताते, एंड्रॉइड ब्राउज़र मान लेगा कि आपका पृष्ठ 980px चौड़ा है (चित्र 2–3). अधिकांश मामलों में, यह बहुत अच्छा काम करता है। हालाँकि, आप प्रारूपित करने जा रहे हैं विशेष रूप से एंड्रॉइड फोन के छोटे आयामों के लिए सामग्री, इसलिए आपको चाहिए मोबाइल ब्राउज़र को हेड element (head element) में व्यूपोर्ट मेटा टैग जोड़कर इसके बारे में बताएं HTML दस्तावेज़ का:

```
<metaname="viewport"content="user-scalable=no,width=device-
widh">
```

डेस्कटॉप ब्राउजर व्यूपोर्ट मेटा टैग को नजरअंदाज करेगा, ताकि आप इसमें शामिल हो सकें यह आपकी साइट के डेस्कटॉप संस्करण के बारे में चिंता किए बिना।

Figure 2-3. Android assumes a normal web page is 980px wide

डेस्कटॉप स्टाइलशीट को दबाकर और अपने व्यूपोर्ट को कॉनफ़िगर करके, आप करेंगे पहले से ही अपने Android उपयोगकर्ताओं को एक बढ़ाया अनुभव दिया है (चित्रा 2–4). वास्तव में उन्हें प्रभावित करें, चलो Android.css स्टाइलशीट का निर्माण शुरू करें।

Figure 2-4. Setting the viewport to the width of the device makes your pages a lot more readable on Android

यदि आप व्यूपोर्ट की चौड़ाई निर्धारित नहीं करते हैं, तो पहले लोड होने पर पृष्ठ को ज़ूम आउट किया जाएगा। यह कहना मुश्किल है कि ज़ूम स्तर क्या होगा क्योंकि एंड्रॉइड ब्राउज़र में एक सेटिंग शामिल है जो उपयोगकर्ताओं को डिफ़ॉल्ट ज़ूम सेट करने की अनुमति देती है। विकल्प दूर, मध्यम (डिफ़ॉल्ट), या बंद हैं। यहाँ तक की यदि आप व्यूपोर्ट की चौड़ाई निर्धारित करते हैं, तो ये उपयोगकर्ता–निर्धारित सेटिंग्स आपके ऐप के ज़ूम स्तर को प्रभावित करेंगी।

Adding the Android CSS

कई यूआई सम्मेलन हैं जो एंड्रॉइड ऐप को एंड्रॉइड ऐप की तरह बनाते हैं। अगले भाग में, हम विशिष्ट title बार जोड़ेंगे, गोल कोनों के साथ सूचियाँ,

उंगली के अनुकूल लिंक जो चमकदार बटन की तरह दिखते हैं, आदि। अपनी पसंद के टेक्स्ट एडिटर के साथ, Android.css नामक एक फाइल बनाएं और उदाहरण 2–2 में दिखाए गए कोड को इसमें जोड़ें, फिर फ़ाइल को उसी निर्देशिका में सहेजें जैसा कि आपका HTML दस्तावेज़ है।

Example 2-2. Setting some general site-wide styles on the HTML body element

```
body {
    background-color: #ddd; /* Background color */
    color: #222;            /* Foreground color used for text */
    font-family: Helvetica;
    font-size: 14px;
    margin: 0;              /* Amount of negative space around the
                               outside of the body */
    padding: 0;             /* Amount of negative space around the
                               inside of the body */
}
```

Android पर सभी text Droid नामक एक कस्टम फ़ॉन्ट का उपयोग करके प्रदान किए गए हैं। Droid फ़ॉन्ट परिवार विशेष रूप से मोबाइल उपकरणों के लिए बनाया गया था, जिसमें उत्कृष्ट वर्ण सेट समर्थन है, और इसमें तीन प्रकार हैं: roid Sans, Droid सैंस मोनो, और ड्रॉइड सेरिफ़। इसलिए, हेवेटिका के एक फ़ॉन्ट परिवार को निर्दिष्ट करने के रूप में हमने यहां किया है, केवल एंड्रॉइड के अलावा अन्य उपकरणों पर प्रभाव पड़ेगा।

अब हेडर div पर हमला करते हैं जिसमें मुख्य होम लिंक (यानी, लोगो लिंक) होता है) और प्राथमिक और द्वितीयक साइट नेविगेशन। पहला चरण लोगो लिंक को प्रारूपित करना है क्लिक करने योग्य title बार के रूप में। Android में निम्न जोड़ें।सीएसएस फ़ाइल:

```
#header h1 {
    margin: 0;
    padding: 0;
}
#header h1 a {
    background-color: #ccc;
    border-bottom: 1px solid #666;
    color: #222;
    display: block;
    font-size: 20px;
    font-weight: bold;
    padding: 10px 0;
    text-align: center;
    text-decoration: none;
}
```

हम प्राथमिक और माध्यमिक नेविगेशन उल ब्लॉकों को प्रारूपिक रूप से प्रारूपित करेंगे, ताकि हम बस कर सकें सामान्य टैग चयनकर्ताओं का उपयोग करें (i.e., #header ul) के विपरीत tag IDs (i.e., #header ul#utiliiy, #header ul#nav):

```
#header ul {
 list-style: none;
 margin: 10px;
 padding: 0;
}
#header ul li a {
 background-color: #FFFFFF;
 border: 1px solid #999999;
 color: #222222;
 display: block;
 font-size: 17px;
 font-weight: bold;
 margin-bottom: -1px;
 padding: 12px 10px;
 text-decoration: none;
}
```

बहुत सरल अब तक, सही? सीएसएस के इस छोटे से के साथ, हम पहले से ही एक बड़ा बना चुके हैं एंड्रॉइड पेज डिजाइन में सुधार (चित्र 2–5). इसके बाद, कुछ पैडिंग जोड़ें सामग्री और साइडबार पाठ को स्क्रीन के किनारे से थोड़ा सा इंडेंट करने के लिए विभाजित करता है (चित्र 2-6):

```
#content, #sidebar {
 padding: 10px;
 }
```

आप सोच रहे होंगे कि हम सामग्री में पैडिंग क्यों जोड़ रहे हैं और साइडबार elements को body के element पर ही विश्व स्तर पर स्थापित करने के बजाय। कारण यह है कि elements को किनारे पर प्रदर्शित करना बहुत आम है बढ़त (इस उदाहरण में हैडर के साथ)। इस वजह से, body या किसी अन्य element में बहुत से लपेटे जाने वाले पैडिंग को लागू किया जाता है दूसरों के लायक होने से ज्यादा परेशानी हो सकती है।

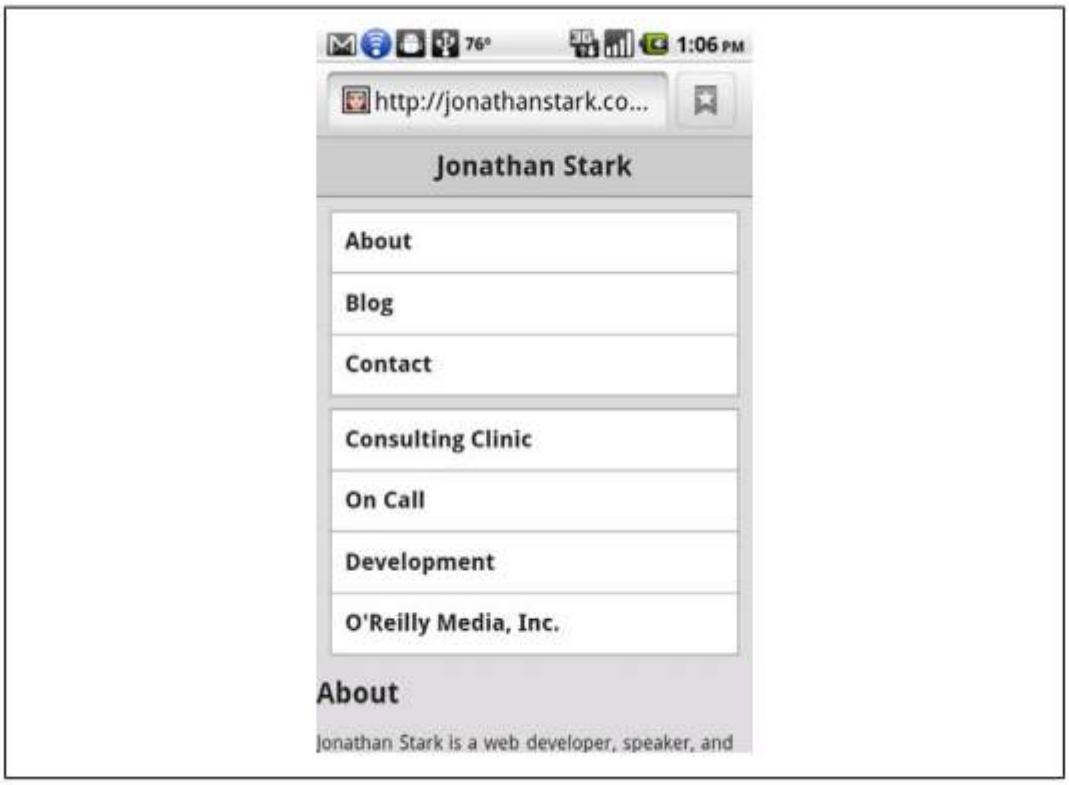


Figure 2-5. A little bit of CSS can go a long way toward enhancing the usability of your Android app

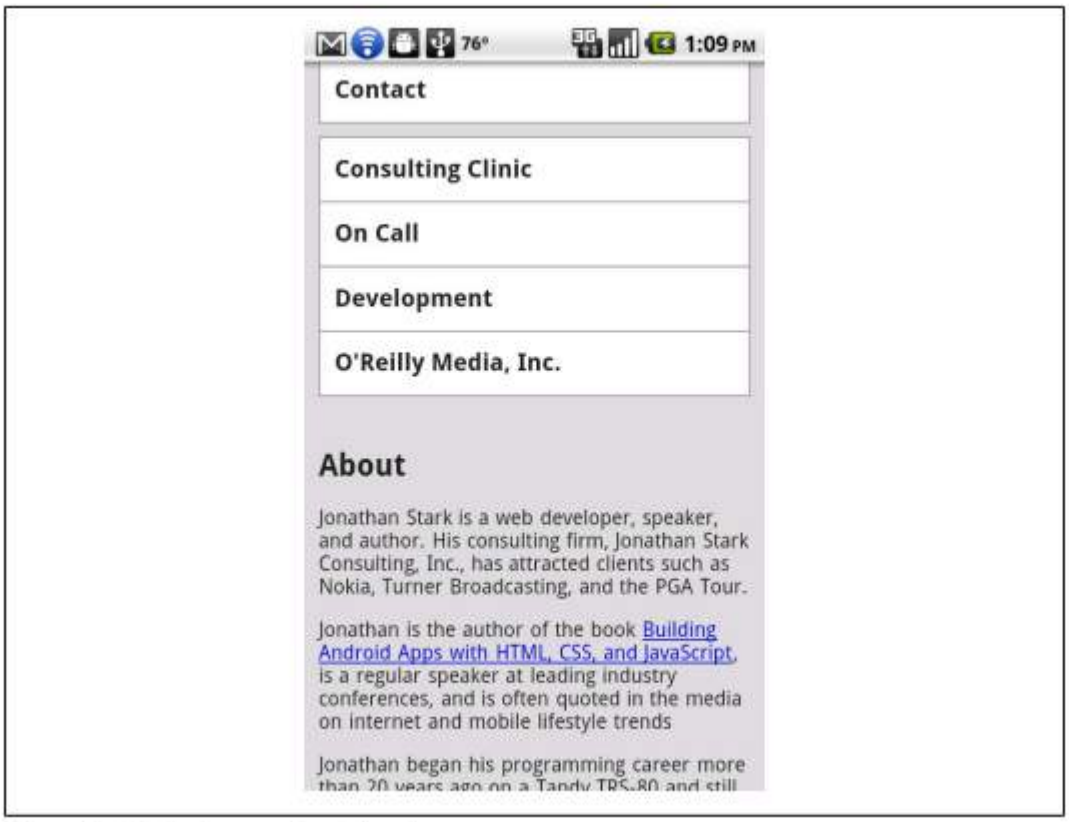

1:09 PM
Contact
Consulting Clinic
On Call
Development
O'Reilly Media, Inc.
About
Jonathan Stark is a web developer, speaker, and author. His consulting firm, Jonathan Stark Consulting, Inc., has attracted clients such as Nokia, Turner Broadcasting, and the PGA Tour.
Jonathan is the author of the book Building Android Apps with HTML, CSS, and JavaScript, is a regular speaker at leading industry conferences, and is often quoted in the media on internet and mobile lifestyle trends
Jonathan began his programming career more

Figure 2-6. Indenting text from the edges

इस पृष्ठ के Footer लेख की सामग्री मूल रूप से नेविगेशन element का एक प्रकार है (पृष्ठ के शीर्ष पर आईडी नेवी के साथ उल element), इसलिए आप पाद लेख को निकाल सकते हैं पृष्ठ के एंड्रॉइड संस्करण से डिस्प्ले को कोई भी सेट नहीं कर सकता, इस प्रकार है:

```
#footer {
 display: none;
}
```

Adding the Android Look and Feel

पृष्ठ के शीर्ष से शुरू होकर, 1–पिक्सेल सफ़ेद ड्रॉप जोड़ें हेडर टेक्स्ट में छाया और बैकग्राउंड के लिए सीएसएस ग्रेडिएंट:

```
#header h1 a {
 text-shadow: 0px 1px 1px #fff;
 background-image: -webkit-gradient(linear, left top, left bottom,
 from(#ccc), to(#999));
```

```
}
```

text-shadow declaration में, बाएं से दाएं पैरामीटर हैं: horizontal ऑफसेट, vertical ऑफसेट, blur, और color । अधिकांश समय, आप यहां दिखाए गए सटीक मानों को अपने पाठ में लागू कर रहे होंगे, क्योंकि यह आमतौर पर एंड्रॉइड पर अच्छा दिखता है, लेकिन यह है text-shadow के साथ प्रयोग करने में मज़ा है क्योंकि यह आपके डिजाइन में एक सूक्ष्म लेकिन परिष्कृत स्पर्श जोड़ सकता है।

अधिकांश ब्राउज़रों पर, 0px की धुंधला त्रिज्या निर्दिष्ट (blur radius) करना ठीक है। तथापि, Android के लिए आवश्यक है कि आप कम से कम 1px का एक धुंधला त्रिज्या निर्दिष्ट करें। यदि आप 0 का एक धब्बा निर्दिष्ट करते हैं, तो पाठ छाया Android उपकरणों पर दिखाई नहीं देगा।

Webkit–gradient लाइन विशेष ध्यान देने योग्य है। यह ब्राउजर को एक निर्देश है कि वह फ्लाई पर एक ग्रेडिएंट इमेज जेनरेट करे। इसलिए, आप कहीं भी सीएसएस ग्रेडिएंट का उपयोग कर सकते हैं आप आम तौर पर एक url () (जैसे, background image, list style image) निर्दिष्ट करेंगे। बाएं से दाएं के पैरामीटर निम्नानुसार हैं: ग्रेडिएंट प्रकार (रैखिक हो सकता है या) रेडियल), ग्रेडिएंट का प्रारंभिक बिंदु (left top, left bottom, right top, or right bottoms हो सकता है), ग्रेडिएंट का अंतिम बिंदु, शुरुआती रंग और अंतिम रंग।

आप चार ग्रेडिएंट स्टार्ट और स्टॉप पॉइंट कॉन्स्टेंट (यानी, लेफ्ट टॉप, लेफ्ट बॉटम, राईट टॉप, या राईट बॉटम) के हॉरिजॉन्टल और वर्टिकल हिस्सों को रिवर्स नहीं कर सकते। दूसरे शब्दों में, ऊपर बाएँ, नीचे बाएँ, ऊपर दाएँ, और नीचे दाईं ओर अमान्य मान हैं। अगला चरण नेविगेशन मेनू में पारंपरिक गोल कोनों को जोड़ना है:

```
#header ul li:first-child a {
 -webkit-border-top-left-radius: 8px;
 -webkit-border-top-right-radius: 8px;
}
#header ul li:last-child a {
 -webkit-border-bottom-left-radius: 8px;
 -webkit-border-bottom-right-radius: 8px;
}
```

जैसा कि आप देख सकते हैं, हम पहली सूची आइटम के शीर्ष दो कोनों पर 8–पिक्सेल त्रिज्या को लागू करने के लिए –webkit–border–radius property के कोने–विशिष्ट संस्करणों का उपयोग कर रहे हैं और अंतिम सूची आइटम के नीचे के दो कोने (चित्र 2–7)

यह अच्छा होगा यदि आप बॉर्डर त्रिज्या को संलग्नक ul पर लागू कर सकते हैं, लेकिन यह काम नहीं करता है। यदि आप इसे आजमाते हैं तो आप देखेंगे कि चाइल्ड लिस्ट आइटमों के class कोनों में होगा ul के गोल कोनों को ओवरफ्लो करें, जिससे प्रभाव को नकारना होगा।

तकनीकी तौर पर, हम त्रिज्या कोनों को उल्टा लगाकर गोल सूची प्रभाव को प्राप्त कर सकते हैं, अगर हम उल की पृष्ठभूमि का रंग सफेद पर सेट करें और इसकी पृष्ठभूमि सेट करें child elements पारदर्शी करने के लिए। हालांकि, जब उपयोगकर्ता सूची में पहले या अंतिम आइटम पर क्लिक करते हैं, तो टैप हाइलाइट स्क्वेर्ड–ऑफ दिखाएगा और यह भयानक लग रहा है। आपकी सबसे अच्छी शर्त है जैसे ही मैंने यहां प्रदर्शन किया है, टैग्स पर गोलाई को लागू करें।

Figure 2-7. Gradients, text shadows, and rounded corners start to transform your web page into a native-looking Android app

की घटनाओं: :first-child और :last-child ऊपर pseudoclasses कहा जाता है। Pseudoclasses CSS चयनकर्ता का एक विशेष प्रकार है जो अनुमति देता है आप ऐसे elements को लक्षित करते हैं जो कुछ निहित प्रासंगिक मानदंडों को पूरा करते हैं। दूसरे शब्दों में, आप विशेषताओं के आधार पर चीजों को स्टाइल कर सकते हैं – जैसे कि जहां वे एक सूची में हैं, चाहे उनका कर्सर फ़ोकस हो, या यदि उन्हें क्लिक किया गया है – बिना अपने मार्कअप को मैन्युअल रूप से अपडेट करने के। के लिये उदाहरण के लिए, ली: :first-child उस पहले अभिभावक का चयन करेगा जो उसके उलटे ul parent का बच्चा है। कोड pseudoclass के बिना, हमें मैन्युअल रूप से एक जोड़ना होगा ब्राउज़र को यह बताने के लिए कि पहले वाला class था, वह पहले वाला था।

Adding Basic Behavior with jQuery

अगला कदम कुछ मूल गतिशील व्यवहार का समर्थन करने के लिए

पृष्ठ पर कुछ जावास्क्रिप्ट जोड़ना है। विशेष रूप से, हम उपयोगकर्ताओं को बड़े सम्मान वाले नेविगेशन को दिखाने और छिपाने की अनुमति देंगे हेडर में सेक्शन ताकि वे केवल तब ही इसे देख सकें जब वे चाहते हैं। इस कार्य को करने के लिए, हम कुछ नए CSS लिखेंगे और मौजूदा HTML में नए CSS को लागू करने के लिए कुछ जावास्क्रिप्ट का उपयोग करेंगे।

सबसे पहले, आइए नए CSS पर एक नज़र डालते हैं। चरण 1 हैडर में उल elements को छिपाने के लिए ताकि उपयोगकर्ता पहले पृष्ठ लोड करने पर वे दिखाई न दें। यदि आप साथ चल रहे हैं , अपना Android.css फ़ाइल खोलें और निम्नलिखित जोड़ें:

```
    #header u1.hide  {
       display: none;
}
```

यह वास्तव में कुछ भी नहीं छिपाएगा जब तक आप छिपने वाले class को उल elements में नहीं जोड़ते हैं (आप कुछ जावास्क्रिप्ट के साथ शीघ्र ही ऐसा करेंगे)। अगला, बटन के लिए शैलियों को परिभाषित करेगा मेनू को दिखाएँ और छिपाएँ। हमने अभी तक बटन के लिए HTML नहीं बनाया है। आपकी जानकारी के लिए, यह इस तरह दिखने वाला है:

```
<div class="leftButton" onclick="toggleMenu()">Menu</div>
```

मैं पेज 28 में jQuery के साथ बेसिक बिहेवियर को जोड़ने वाले अनुभाग में विस्तार से बटन HTML का वर्णन करूँगा, इसलिए अपने HTML फ़ाइल में कोड की पूर्ववर्ती पंक्ति न जोड़ें। समझने की महत्वपूर्ण बात यह है कि यह क्लास लेफ्टबटन के साथ एक div है और यह हेडर में रहने वाला है।

यहां बटन के लिए सीएसएस Style है (आप आगे बढ़ सकते हैं और इसे android.css फ़ाइल में जोड़ सकते हैं।):

```
#header div.leftButton {
    position: absolute;❶
    top: 7px;
    left: 6px;
    height: 30px;❷
    font-weight: bold;❸
    text-align: center;
    color: white;
    text-shadow: rgba❹(0,0,0,0.6) 0px -1px 1px;
    line-height: 28px;❺
    border-width: 0 8px 0 8px;❻
    -webkit-border-image: url(images/button.png) 0 8 0 8;❼
}
```

इस अध्याय में प्रयुक्त ग्राफिक्स के लिए, आप http://examples.oreilly.com/catalog/9781449383268 उदाहरणों से उदाहरण फ़ाइलों को डाउनलोड कर सकते हैं। इन प्रतियों को उस निर्देशिका के नीचे एक उपनिर्देशिका की छवियों में रखें जिसमें आपका HTML दस्तावेज़ है (आपको शायद चित्र निर्देशिका बनाने की आवश्यकता होगी)। हम अध्याय 4 में विस्तार से jQTouch के बारे मे बात करेंगे।

- इसे ऊपर से लेते हुए, div से निकालने के लिए स्थिति को पूर्ण पर सेट करें
- दस्तावेज़ प्रवाह। यह आपको इसके शीर्ष और बाएँ पिक्सेल निर्देशांक को सेट करने की अनुमति देता है।
- ऊँचाई को 30px पर सेट करें ताकि यह आसानी से टैप करने के लिए पर्याप्त बड़ा हो।
- टेक्स्ट को बोल्ड, थोड़ा ड्रॉप शैडो के साथ सफ़ेद करें और बॉक्स में केंद्रित करें। CSS में, आरजीबी फ़ंक्शन आमतौर पर

उपयोग किए जाने वाले परिचित हेक्स नोटेशन का एक विकल्प है

रंग निर्दिष्ट करने के लिए (e.g., #FFFFFF). आरजीबी (255, 255,255) और rgb(100%, 100%, 100%) दोनों के समान है #FFFFFF. हाल ही में, आरजीबीए () फ़ंक्शन पेश किया गया है, जो आपको एक चौथा पैरामीटर निर्दिष्ट करने की अनुमति देता है जो अल्फा मान को परिभाषित करता है (यानी, रंग की अस्पष्टता)। स्वीकार्य मूल्यों की सीमा 0 से 1 है, जहां 0 पूरी तरह से पारदर्शी है और 1 पूरी तरह से अपारदर्शी है; 0 और 1 के बीच दशमलव मान प्रदान किया जाएगा पारदर्शी।

- line heigh declaration पाठ को बॉक्स में लंबवत रूप से नीचे ले जाती है, इसलिए यह शीर्ष सीमा के विरुद्ध फ्लश नहीं है।
- बॉर्डर–चौड़ाई और –webkit– बॉर्डर–इमेज लाइनों को थोड़ा स्पष्टीकरण की आवश्यकता है। इन एक साथ दो गुण आपको एक element के सीमा क्षेत्र में एक एकल छवि के कुछ हिस्सों को असाइन करने की अनुमति देते हैं। यदि बॉक्स आकार बदलता है क्योंकि पाठ बढ़ता है या घटता है, तो इसे समायोजित करने के लिए सीमा की छवि खिंच जाएगी। यह वास्तव में बहुत अच्छी बात है क्योंकि इसका मतलब है कम छवियां, कम काम, कम बैंडविड्थ और कम लोड समय।

  सीमा–चौड़ाई रेखा ब्राउज़र को शीर्ष पर 0 चौड़ाई सीमा, दाईं ओर 8px सीमा, नीचे की ओर 0 चौड़ाई सीमा और 8px चौड़ाई सीमा लागू करने के लिए ब्राउज़र को बताती है बाएं (यानी, चार पैरामीटर बॉक्स के शीर्ष पर शुरू होते हैं और दक्षिणावर्त चारों ओर अपना काम करते हैं)। सीमा के लिए आपको रंग या शैली निर्दिष्ट करने की आवश्यकता नहीं है।
- सीमा चौड़ाई के साथ, आप सीमा छवि लागू कर सकते हैं।

बाएं से दाएं पांच पैरामीटर हैं: छवि का URL, शीर्ष चौड़ाई, दाईं चौड़ाई, नीचे की चौड़ाई, और बाईं चौड़ाई (फिर से, ऊपर से दक्षिणावर्त)। URL निरपेक्ष हो सकता है (http://example.com/myBorderImage.png) या रिश्तेदार। सापेक्ष पथ आधारित हैं स्टाइलशीट के स्थान पर, HTML पृष्ठ नहीं है जिसमें स्टाइलशीट शामिल है।

- जब मैंने पहली बार बॉर्डर इमेज प्रॉपर्टी का सामना किया, तो मुझे यह अजीब लगा कि मुझे बॉर्डर की चौड़ाई निर्दिष्ट करनी थी जब मैंने पहले ही ऐसा कर लिया था border width की property के साथ। कुछ दर्दनाक परीक्षण और त्रुटि के बाद, मुझे पता चला कि border –image property में चौड़ाई सीमा चौड़ाई नहीं है; वे छवि से टुकड़ा करने की चौड़ाई हैं। लेना
  एक उदाहरण के रूप में दाईं सीमा, मैं ब्राउज़र को छवि के बाईं ओर 8px लेने के लिए और उन्हें सही सीमा पर लागू करने के लिए कह रहा हूं, जो कि यह भी होता है एक 8px चौड़ाई।

  कुछ तर्कहीन करना संभव है जैसे कि सही 4 को लागू करना
  एक छवि की बॉर्डर जो कि 20px चौड़ी है। इस काम को ठीक से करने के लिए, आपको वेबकिट–बॉर्डरिमेज के वैकल्पिक मापदंडों का उपयोग करना होगा जो छवि को निर्देश देता है कि उपलब्ध स्लाइस के साथ क्या करें सीमा स्थान (दोहराने, खिंचाव, गोल, आदि)। तीन साल की कोशिश में, मैं ऐसा करने के लिए किसी भी कारण के साथ आने में विफल रहा हूं, इसलिए मैं इस भ्रामक और अव्यवहारिक विकल्प का वर्णन करते हुए

यहां जगह बर्बाद नहीं करूंगा।

ठीक है, कुछ जावास्क्रिप्ट के लिए समय। आपके द्वारा लिखने वाले जावास्क्रिप्ट की तैयारी में, आपको jquery.js को शामिल करने के लिए अपने HTML दस्तावेज़ को अपडेट करने की आवश्यकता है और Android.js। इन लाइनों को अपने HTML डॉक्यूमेंट के हेड सेक्शन में जोड़ें:

```
<script type="text/javascript"  src="jquery.js"></sclipt>
<script type="text/javascript" src="android.js"></sclipt>
```

http:// jQuery.com पर के डाउनलोड, प्रलेखन और ट्यूटोरियल उपलब्ध हैं: JQuery का उपयोग करने के लिए, आपको इसे वहां से डाउनलोड करना होगा, आपके द्वारा डाउनलोड की गई फ़ाइल का नाम बदलें (जैसे jquery–1.3.2.min.js) को jQuery.js, और अपने HTML के समान निर्देशिका में इसकी एक प्रति दस्तावेज डालें।

Android.js में जावास्क्रिप्ट का प्राथमिक कर्तव्य उपयोगकर्ताओं को दिखाने और छिपाने की अनुमति देता है nav मेनू। Android.js नामक फ़ाइल में निम्नलिखित जावास्क्रिप्ट कॉपी करें और इसे में सहेजें HTML फ़ाइल के समान फ़ोल्डर:

```
if (window.innerWidth && window.innerWidth <= 480) { ❶
    $(document).ready(function(){ ❷
        $('#header ul').addClass('hide'); ❸
        $('#header').append('<div class="leftButton"
            onclick="toggleMenu()">Menu</div>'); ❹
    });
    function toggleMenu() {
        $('#header ul').toggleClass('hide'); ❺
        $('#header .leftButton').toggleClass('pressed'); ❻
    }
}
```

1. कोड के पूरे ब्लॉक को एक निश्चित विवरण में लपेटा गया है

जो सुनिश्चित करने के लिए जांच करता है विंडो ऑब्जेक्ट की आंतरिक property मौजूद है (यह कुछ संस्करणों में मौजूद नहीं है इंटरनेट एक्सप्लोरर का) और वह चौड़ाई 480px (एक उचित) से कम या बराबर है अधिकतम फोन के लिए अधिकतम चौड़ाई)। इस पंक्ति को जोड़कर, हम यह सुनिश्चित करते हैं कि कोड केवल तभी निष्पादित होता है जब उपयोगकर्ता किसी विशिष्ट Android फ़ोन के साथ पृष्ठ ब्राउज़ कर रहा हो या कुछ इसी तरह के आकार के उपकरण।

यदि आप डेस्कटॉप संस्करण का उपयोग करके अपने Android वेब पृष्ठों का परीक्षण कर रहे हैं क्या आपके पास वेबसाइट नहीं है? पृष्ठ 13 पर, यदि आपके ब्राउज़र की विंडो की चौड़ाई बहुत अधिक है, तो यहां कथन विफल हो जाएगा । मैन्युअल रूप से संभव के रूप में संकीर्ण होने के लिए अपनी window का आकार बदलें पृष्ठ ताज़ा करें।

2. यहां हमारे पास तथाकथित दस्तावेज़ तैयार फ़ंक्शन है। यदि आप jQuery के लिए नए हैं, यह थोड़ा डराने वाला हो सकता है, और मैं स्वीकार करता हूं कि मुझे याद करने में थोड़ा समय लगा syntax। हालाँकि, यह समय इसे स्मृति में ले जाने के लायक है, क्योंकि आप
   इसका भरपूर उपयोग करें। दस्तावेज़ तैयार फ़ंक्शन मूल रूप से कहता है, जब दस्तावेज़ तैयार है, इस कोड को चलाएं। एक सेकंड में यह महत्वपूर्ण क्यों है।

3. यह विशिष्ट jQuery कोड है जो हेडर में il का चयन करके और जोड़ना शुरू करता है सीएसएस class उन्हें छिपाने के लिए। याद रखें, छिपाना वह चयनकर्ता है जिसका उपयोग

हमने ऊपर सीएसएस में किया था।
इस लाइन को निष्पादित करने पर शुद्ध प्रभाव हेडर ul element को अच्छी तरह से छिपाना है।

अगर हमने इस लाइन को दस्तावेज़ तैयार फ़ंक्शन में नहीं लपेटा होता, तो यह सबसे अधिक संभावना है कि uls समाप्त होने से पहले ही निष्पादित हो जाएंगे लोड हो रहा है। इसका मतलब यह है कि जावास्क्रिप्ट लोड होगा, और यह लाइन होगी विफल क्योंकि uls अभी तक मौजूद नहीं होगा। फिर, पेज लोड होता रहेगा, uls दिखाई देगा, और आप अपना स्क्रैचिंग करेंगे Head (या अपने कीबोर्ड को तोड़ते हुए), सोच रहा था कि जावास्क्रिप्ट क्यों काम नहीं कर रहा था।

4. यहां हम शीर्ष लेख के लिए एक बटन जोड़ते हैं, जो उपयोगकर्ता को दिखाने की अनुमति देगा और मेनू छिपाएं (चित्र 2–8). इसमें एक class है जो हमने लिखा सीएसएस से मेल खाता है पहले के लिए .left buttton, और इसमें एक ऑनक्लिक हैंडलर है जो फ़ंक्शन को कॉल करता है TogleMenu() जो आगे आता है।

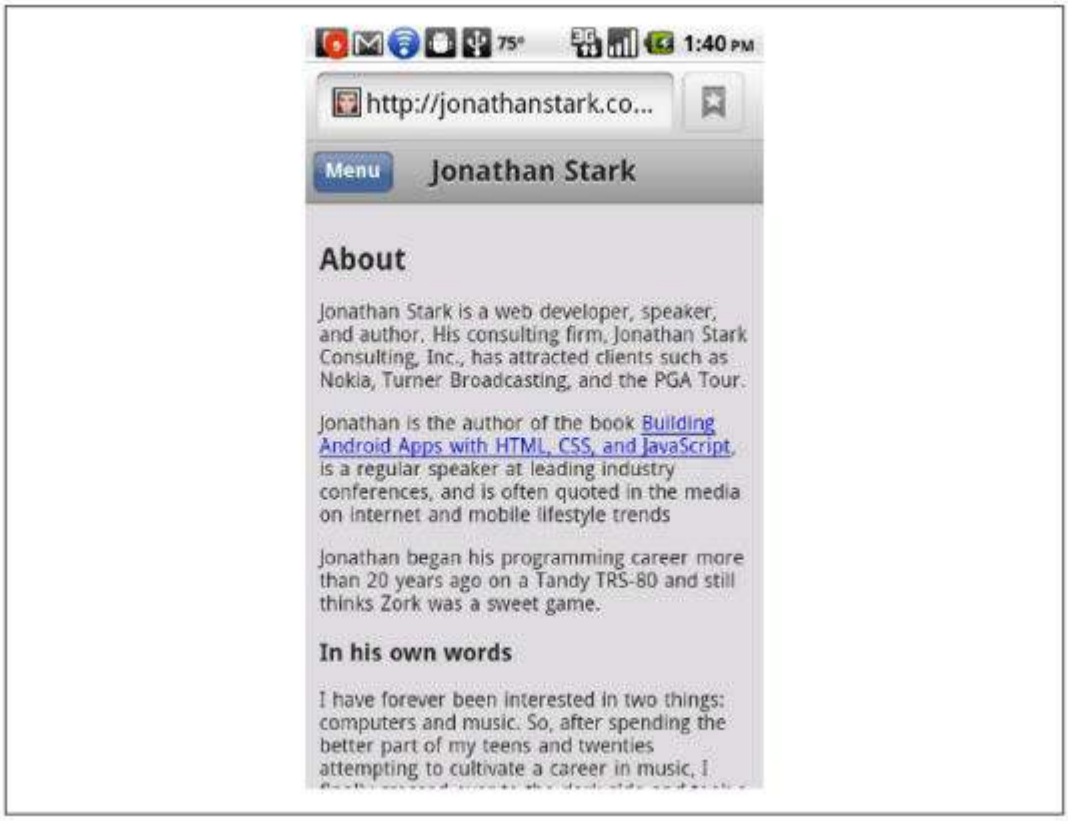


Figure 2-8. The Menu button has been added to the toolbar dynamically using jQuery

5. TogleMenu () फ़ंक्शन jQuery के toggleClass () फ़ंक्शन को जोड़ने या निकालने के लिए फ़ंक्शन का उपयोग करता है चयनित ऑब्जेक्ट के लिए निर्दिष्ट class। इस लाइन पर, हम पर छिपा class टॉगल करें हेडर uls।

6. यहां, हम हेडर लेफ्टबटन पर प्रेस की गई क्लास को टॉगल करते हैं। यह सोचने के लिए आओ, हमने अभी तक दबाए गए class के लिए सीएसएस नहीं लिखा है, तो चलो ऐसा करते हैं अभी Android.css पर वापस जाएं और निम्नलिखित डालें:

```
#header div.pressed {
-webkit-border-image:url(images/button_c1icked.png) 0 8 0 8;
}
```

जैसा कि आप देख सकते हैं, हम बस बटन बॉर्डर के लिए एक

अलग छवि निर्दिष्ट कर रहे हैं (यह थोड़ा गहरा होता है)। यह बटन के लिए दो–अवस्था प्रभाव जोड़ देगा उपयोगकर्ता को यह स्पष्ट कर दें कि बटन मेनू को दिखा और छिपा सकता है (चित्र 2–9). चित्र 2–10 पृष्ठ का एक फैला हुआ दृश्य दिखाता है जो दोनों मेनू को दिखाता है और कुछ पाठ।

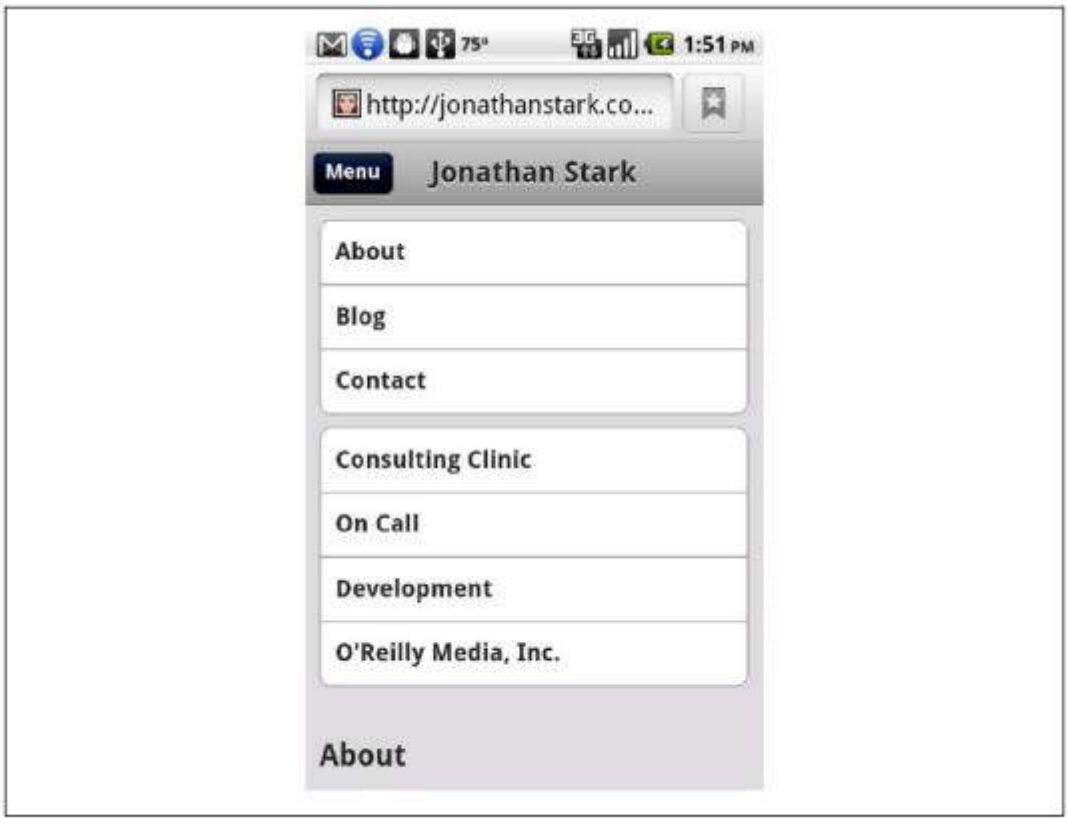


Figure 2-9. The Menu button displays darker when it has been pressed

Figure 2-10. A tall view of the completed basic Android CSS

What You've Learned

इस अध्याय में, हमने एक मौजूदा वेब पेज को और अधिक परिवर्तित करने की मूल बातें कवर कीं Android के अनुकूल प्रारूप। हम भी दिखाने के लिए और छिपाने के लिए गतिशील HTML का एक सा इस्तेमाल किया नेविगेशन पैनल। अगले अध्याय में, हम खोज करते हुए इन उदाहरणों का निर्माण करेंगे

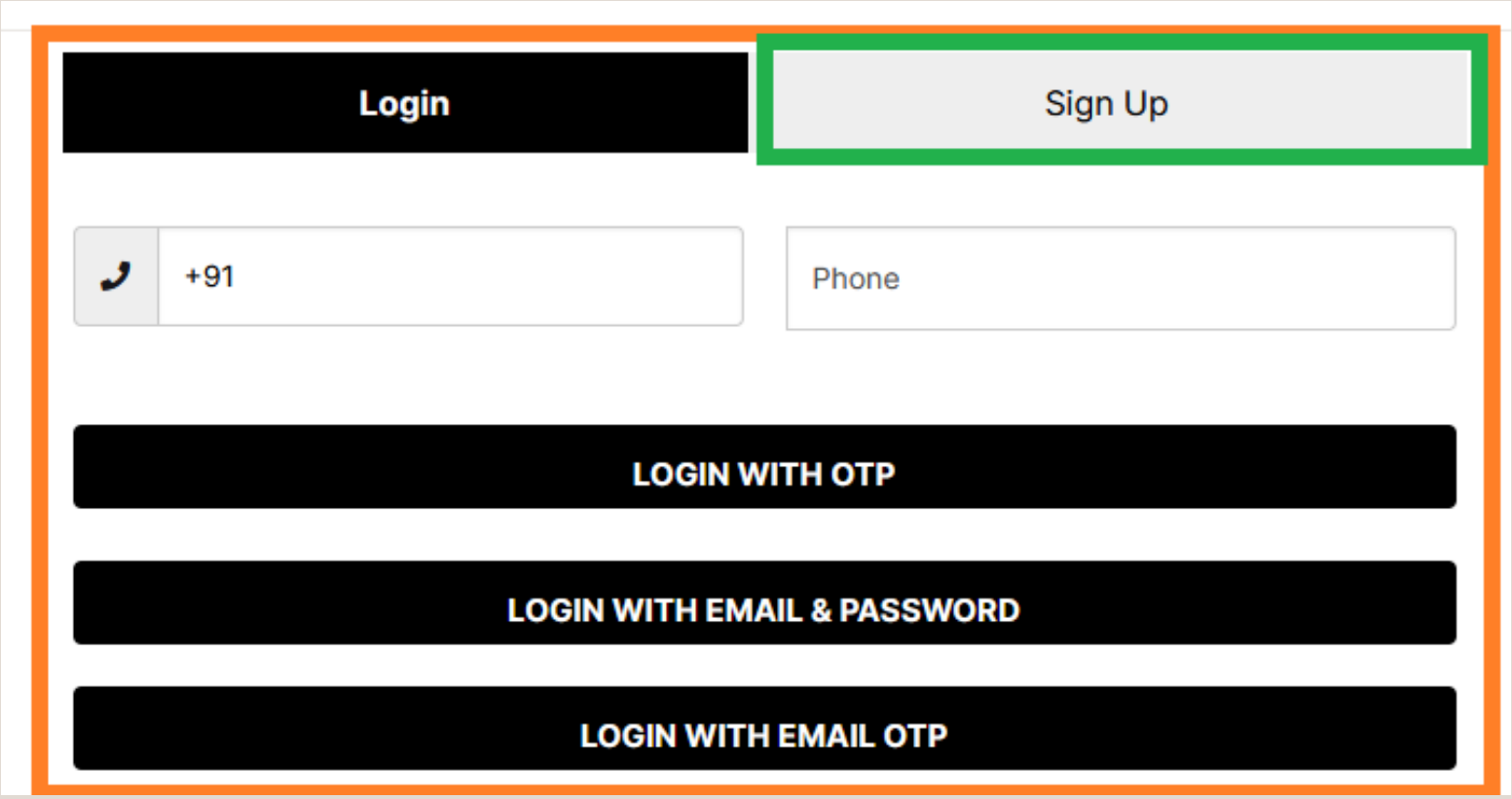
कुछ और उन्नत जावास्क्रिप्ट अवधारणाओं; विशेष रूप से, कुछ अजाक्स हम चर्चा करेंग।

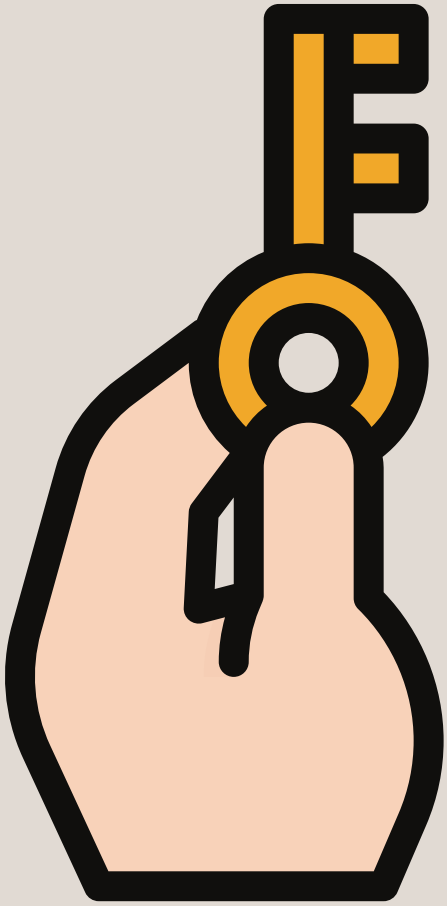
HOW TO GET COMPLETE PDF EBOOK

Pay Paytm

कोई भी **2 Ebooks** खरीदने पर **20%** डिस्काउंट- **India20**

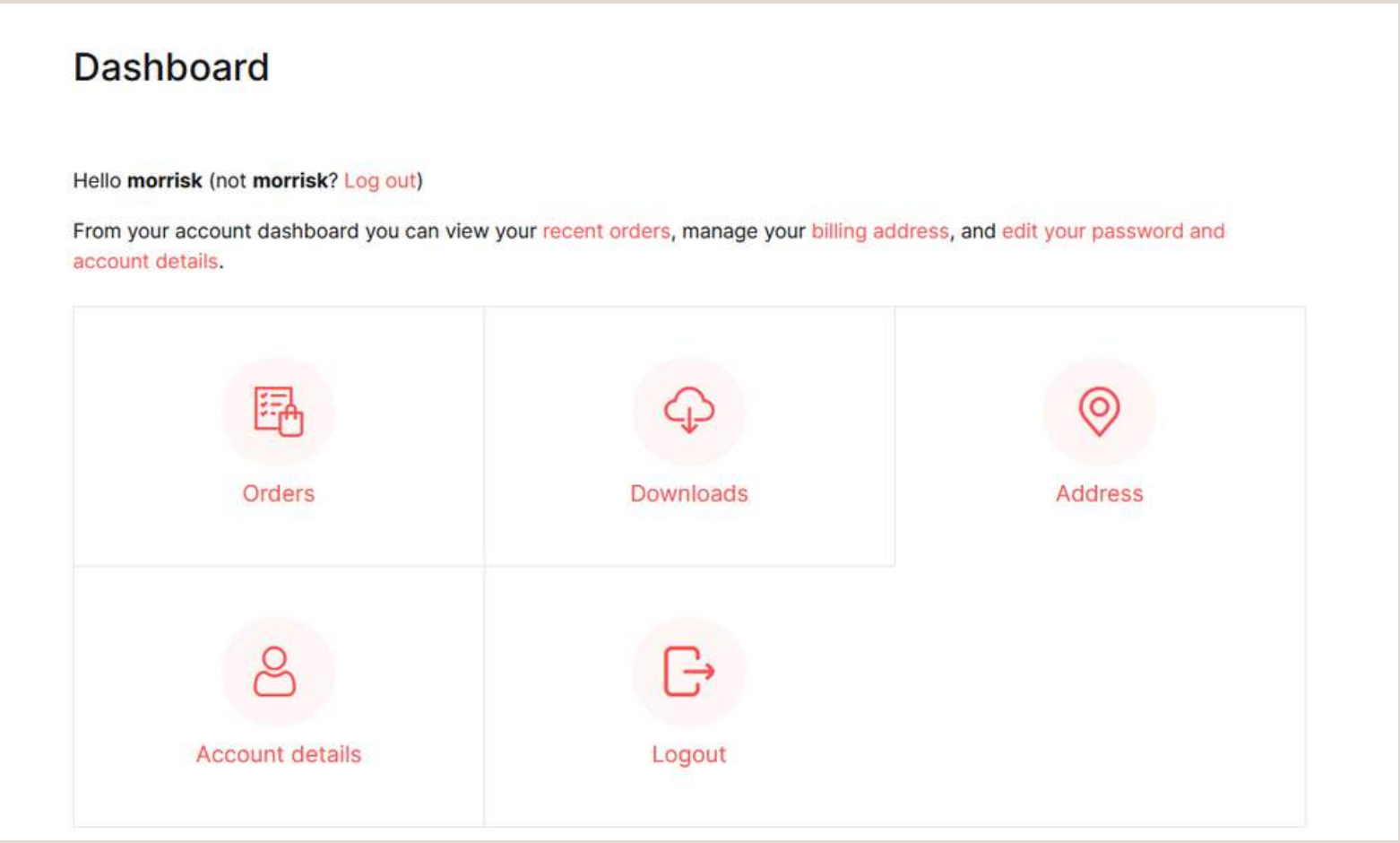
Buy Now पर क्लिक करने पर आपको **चेकआउट पेज** पर लॉगिन फॉर्म पर पहिले ईमेल से लॉगिन करना होगा।





Login Succesfull के बाद आपको पेमेंट करके Ebooks वही से डाउनलोड कर सकते है।

उसके बाद आप **My Account** पेज पर जाकर **Download** में Ebooks डाउनलोड कर सकते है।



Thank you